ESL-TR-87-35

ATMOSPHERIC PHOTOCHEMICAL MODELING OF TURBINE ENGINE FUELS AND EXHAUST, PHASE II, COMPUTER MODEL DEVELOPMENT, VOLUME II OF II

DR. W.P.O. CARTER, A.M. WINER, R. ATKINSON,
S.E HEFFRON, M.P. POE, DR. M.A. GOODMAN

UNIVERSITY OF CALIFORNIA
STATEWIDE AIR POLLUTION RESEARCH CENTER
UNIVERSITY OF CALIFORNIA
RIVERSIDE CA 92521

MAY 1988

FINAL REPORT

OCTOBER 1984 - DECEMBER 1986

ENGINEERING & SERVICES LABORATORY
AIR FORCE ENGINEERING & SERVICES CENTER
TYNDALL AIR FORCE BASE, FLORIDA 32403

NOTICE

PLEASE DO NOT REQUEST COPIES OF THIS REPORT FROM HQ AFESC/RD (ENGINEERING AND SERVICES LABORATORY). ADDITIONAL COPIES MAY BE PURCHASED FROM:

NATIONAL TECHNICAL INFORMATION SERVICE
5285 PORT ROYAL ROAD
SPRINGFIELD, VIRGINIA 22161

FEDERAL GOVERNMENT AGENCIES AND THEIR CONTRACTORS REGISTERED WITH DEFENSE TECHNICAL INFORMATION CENTER SHOULD DIRECT REQUESTS FOR COPIES OF THIS REPORT TO:

DEFENSE TECHNICAL INFORMATION CENTER
CAMERON STATION
ALEXANDRIA, VIRGINIA 22314

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

ADA198691

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release Distribution unlimited
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S) ESL-TR-87-35
6a. NAME OF PERFORMING ORGANIZATION University of California Riverside CA 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION Air Force Engineering and Services Center
6c. ADDRESS (City, State, and ZIP Code) Statewide Air Pollution Research Center University of California Riverside CA 92521	7b. ADDRESS (City, State, and ZIP Code) HQ AFESC/RDVS Tyndall AFB FL 32403
8a. NAME OF FUNDING/SPONSORING ORGANIZATION 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER F08635-83-0278
8c. ADDRESS (City, State, and ZIP Code)	10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO	PROJECT NO.	TASK NO	WORK UNIT ACCESSION NO
62601F	1900	20	40

11. TITLE (Include Security Classification)
Atmospheric Photochemical Modeling of Turbine Engine Fuels and Exhaust, Phase II, Computer Model Development, Volume II of II

12. PERSONAL AUTHOR(S) Dr William P.O. Carter, Arthur M. Winer, Roger Atkinson, Ms Susan E. Heffron, Ms Minn P. Poe, and Dr Mark A. Goodman

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Final	FROM Oct 84 TO Dec 86	May 1988	201

16. SUPPLEMENTARY NOTATION
Availability of this report is specified on reverse of front cover.

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)	
FIELD	GROUP	SUB-GROUP		
04	01		Photochmeical Models	Atmospheric Reactivity
07	04		Computer Model Software	Airshed Models,
			Air Pollution	ROG Emissions

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

This User's Manual documents how to use an atmospheric photochemical reactivity modeling software system which has been developed for the U.S. Air Force under this contract. The modeling software can be used to conduct "floating box" type airshed model simulations to determine the air quality impacts of many types of reactive organic gases (ROG) emitted under a variety of air pollution scenarios. This software is written in FORTRAN and

(Continued)

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT	21. ABSTRACT SECURITY CLASSIFICATION
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE (Include Area Code)	22c. OFFICE SYMBOL
SURENDRA B. JOSHI	(904) 283-4235	RDVS

DD Form 1473, JUN 86 *Previous editions are obsolete.* SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

20. ABSTRACT (concluded)

can run on DEC VAX and CDC Cyber computer systems and the User's Manual provides guidance for its use on either system.

The photochemical reactivity modeling system includes: (1) an interactive "User Interface" program which can be used to define conditions of air pollution scenarios, to specify compositions of background, aloft, and emitted pollutants, to submit model calculations, and to tabulate and plot the results; (2) a "Model Integration" program which carries out the model simulations, (3) a "Plotting" program which can be used to plot or tabulate the results outside the "User Interface", and (4) a "Model Preparation program which translates data files, giving the chemical mechanism in a form which can be used by the integration program. The manual documents primarily the User Interface and the Plotting programs, since these are all that are required to use this model once it is installed. The manual also includes instructions for installing the software on VAX and Cyber computers, and an overview of the other programs and how they are related. Files for representative pollution scenarios and pollution mixtures are included with this software, and examples of how these can be used in reactivity assessment calculations are discussed in this manual.

Air pollution scenarios in this model are represented by a single well-mixed air mass which can have variable mixing heights, separate emissions schedules for NO_x, for base case ROG, and test mixture ROG emissions, variable light intensities, entrainment of pollutants from aloft as the inversion height rises, and a number of other characteristics, giving it a high degree of flexibility for a single box model. It is capable of representing the scenarios used in present EPA city-specific EKMA models.

This model incorporates a detailed and extensively tested state-of-the-art chemical mechanism, whose development and testing is documented in a separate report for this contract. This mechanism includes separate representations for a wide variety of classes of ROG species, including $>C_9$ alkanes, $>C_6$ alkenes, tetralin, naphthalenes, furan, thiophene, and pyrrole. The latter have not been included in any previous atmospheric photochemical mechanisms. These species are included so this model can be used to calculate the atmospheric reactivities of the present and potential future components of turbine engine (jet) fuels and jet exhaust, which is the principal purpose for which this model was developed. It can also be used for a variety of other types of ROG emissions.

Although this manual discusses the use of this modeling system for the chemical mechanism presently implemented with it, the software can be used with any other reasonable chemical mechanism. However, a discussion of the use of this software with other mechanisms is beyond the scope of this document.

Computer tapes containing copies of this software, the data files needed to implement the chemical mechanism, and the sample calculations discussed in this document are available from the authors, upon request.

PREFACE

This report was prepared by the Statewide Air Pollution Research Center (SAPRC) of the University of California, Riverside, California 92521, under Contract No. F08635-80-C-0359, with the Air Force Engineering and Services Center, Air Force Engineering and Services Laboratory (AFESC/RDVS), Tyndall Air Force Base, Florida 32403.

This report describes the second phase of a two-phase program aimed at developing experimentally tested models for the atmospheric reactions of turbine engine fuels. This phase consisted primarily of the model development and testing and software development, although additional environmental chamber experiments were conducted for several jet engine exhaust constituents.

Volume I of this technical report describes the model development, validation and testing. A user's manual and necessary documentation is being submitted separately as Volume II.

This work was carried out between October 1984 and December 31, 1986 under the direction of Dr William P. L. Carter and Dr Arthur M. Winer, Co-principal Investigators, and Dr Roger Atkinson, Program Manager. The principal research staff on this program was Ms Susan E. Heffron, Ms Minn P. Poe, and Dr Mark A. Goodman. The environmental chamber experiments were carried out with the assistance of Ms Li Li N. Parker and Mr William D. Long. Assistance in preparation of this report was provided by Ms Christy J. LaClaire and Ms Diane L. Skaggs.

Mr Surendra Joshi, AFESC/RDVS, was Project Officer for Phase II of this contract.

This report has been reviewed by the Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nationals.

This report has been reviewed and is approved for public release.

SURENDRA B. JOSHI
Project Officer

THOMAS J. WALKER, Lt Col, USAF, BSC
Chief, Environics Division

KENNETH T. DENBLEYKER, Maj, USAF
Chief, Environmental Sciences Branch

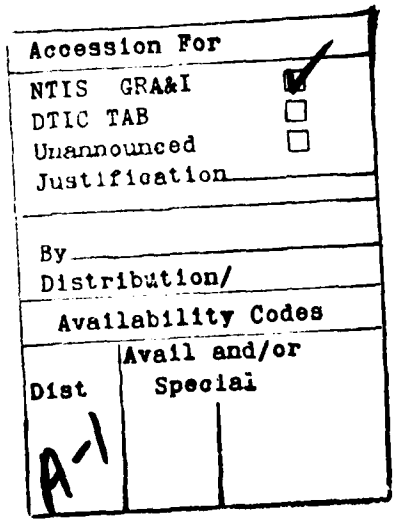
LAWRENCE D. HOKANSON, Colonel, USAF
Director, Engineering and Services Laboratory

(The reverse of this page is blank)

TABLE OF CONTENTS

TABLE OF CONTENTS
(CONTINUED)

TABLE OF CONTENTS
(CONCLUDED)



LIST OF FIGURES

LIST OF TABLES

LIST OF TABLES
(CONCLUDED)

SECTION I

INTRODUCTION

A. OBJECTIVE

This User's Manual has been developed as a supporting document for the Final Technical Report to Phase II of USAF Contract No. F08635-83-0278, entitled Atmospheric Photochemical Modeling of Turbine Engine Fuels, Phase II: Computer Model Development. The overall objective of this program was to develop an experimentally validated chemical kinetic computer model which can be used by the United States Air Force to predict atmospheric impacts of ground level emissions of present and future turbine engine fuels and jet exhausts.

The atmospheric photochemical reactivity modeling system described in this document consists of a series of computer programs and data files which can be used to conduct airshed model simulations. The system was developed with the specific objective of allowing the Air Force to assess the relative atmospheric reactivities of turbine engine fuels and of jet engine exhaust, and to assess whether future changes in the chemical composition of such fuels or exhaust might have an adverse impact on air quality in areas where their emissions are significant.

B. BACKGROUND

Airshed models can vary greatly in complexity, ranging from simple "rollback" models which do not take into account the nature of the specific airshed involved or the chemical composition of the reactive organics emitted, to complex, multicelled "grid" models, which have huge data requirements and also require extensive computational resources. Except for the simplest versions, all airshed models have two major components: (1) the representation of the chemistry (i.e., the chemical nature of the emitted pollutants and the reactions they undergo), and (2) the representation of the physical conditions associated with the airshed into which the emissions are occurring.

In this document, the representation of the physical conditions in an airshed will be referred to as the model "scenario." Scenarios include such factors as atmospheric mixing heights, transport of pollutants, how emissions change with time and location, initial conditions, temperature, and relative humidity. Because of computer constraints, useful airshed models must incorporate extensive simplifications and approximations in at least one, if not both, of the two major components. The nature of such simplifications determines the applications for which a given model is appropriate. For example, a model with extremely detailed representation of the chemistry, but which represents the entire airshed as a single "box," would not be particularly useful in assessing effects of moving emissions sources from one area of the airshed to another. On the other hand, a model with a highly simplified representation of the chemistry would not be useful in assessing effects of changes in pollutant chemical composition, such as might occur, for example, if there were major changes in the formulation of fuels emitted into particular airsheds.

As indicated above, the essential purpose of the modeling system discussed in this manual is the estimation of the effects of changing the chemical composition of emitted mixtures on resulting photochemical air pollution levels. Therefore, this model is of the type which employs a highly detailed representation of the chemical transformations which occur in photochemical air pollution systems. The chemical mechanism incorporated in this model represents the atmospheric reactions of over 20 types of emitted reactive organic species. These include species representative of the major anthropogenic emissions into urban airsheds, and components of present Air Force turbine engine fuels, and several potential future fuel impurities.

The mechanism in this model is based on comprehensive evaluations of our current knowledge of atmospheric chemistry and of the atmospheric reaction mechanisms of organic compounds. As such, it represents the current state of the art in this area. The mechanism has been tested against the results of over 400 environmental chamber experiments conducted at SAPRC and the University of North Carolina, including experiments employing mixtures representing Air Force fuels and jet engine exhaust. The most recent development of this mechanism was funded by both

the USAF and the U.S. Environmental Protection Agency (EPA), but it builds upon previously published photochemical kinetic models developed by atmospheric scientists at SAPRC and elsewhere, under funding from a variety of sources. The development and testing of this chemical mechanism is documented in the Final Technical Report to Phase II of this project (Reference 1).

C. SCOPE

Because this model incorporates such a detailed representation of the chemical transformations, it contains a more approximate representation of the physical characteristics of the airsheds whose air quality are being simulated. In particular, the formulation of this model is based on the "floating box" approach. In this approach, a single well-mixed air mass of variable area (determined by the mixing height which in general changes with time), moves from place to place, receiving emissions from the areas over which it moves. This formulation neglects exchange of pollutants from nearby air masses, other than input of pollutants aloft (if any) which are entrained into the air mass as the inversion height rises. It also assumes that mixing of emitted pollutants between the ground and the inversion height is essentially instantaneous, and that a large air mass can retain its integrity throughout the day. The latter assumption in effect neglects the possibility of wind shear and other factors which might result in much more complex mixing patterns.

This formulation does, however, permit a relatively large degree of flexibility in representing conditions of particular airsheds, since it permits varying both the composition and the rates of pollutant emission with time. It can represent, for example, an air mass moving from an industrial area to a nonindustrial one, or from a rural area to an Air Force base to another rural area, etc. It also permits variation in meteorological factors such as changes in inversion height with time and specification of dilution, temperature, and light intensity.

Although the types of pollution scenarios which can be represented by this model are simplifications relative to real conditions, the model allows a wide degree of flexibility in the ranges of conditions which can

be represented, and permits some customization of the conditions of particular interest to the user. This allows the effects of changes in the composition of emissions to be assessed under a range of conditions which can represent, at least semiquantitatively, the range of conditions which can occur in real airsheds.

SECTION II

OVERVIEW OF THE MODEL

A. OVERVIEW OF APPLICATIONS

The principal application for this modeling system is to assess the relative reactivities of atmospheric mixtures of reactive volatile organic compounds of interest to the USAF. The "atmospheric reactivity" of a "test mixture" (or test compound) under the conditions of a given pollution "scenario" is defined as the effect on some measure of air quality, caused by the emission of given amounts of the test mixture when it is emitted into an air basin under a given set of conditions defined by the scenario. The reactivity of the test compound or mixture will in general depend on the conditions of the scenario (Reference 2). Thus, these conditions should be represented as accurately as possible. In particular, the air parcel of interest may already be polluted by emissions from other sources and the presence of these other "base case" or "background" pollutants can affect the calculated reactivities of the test substances (Reference 2). This reactivity assessment modeling system allows the presence of initial, "background" pollutants, and emissions of "base case" (i.e., "nontest" pollutants to be specified as parts of the overall airshed scenario.

This modeling system can be used to assess the reactivities of a variety of substances and test mixtures, primarily turbine engine fuels and jet exhaust of interest to the Air Force. However, it can also be used to assess the atmospheric reactivities of many other mixtures of hydrocarbons and oxygenated species. These include motor vehicle fuel and exhausts, hydrocarbon solvents, and other types of chemical compounds or mixtures emitted into the atmosphere from military, government or private sector activities. Section III lists the specific types of compounds and mixtures for which the present version of this atmospheric photochemical reactivity model can be applied. It also discusses how the reactions of the various types of individual compounds are represented in the present version of the chemical model.

The components of the model when used for estimating atmospheric reactivities are shown in Figure 1. The major inputs to this model are the model for the chemical transformations, the input data defining the conditions of the airshed or pollution episode of interest (the scenario), the chemical composition of the test substance whose reactivity is being assessed, and the total amount of test substance emitted. The development of the model for the chemical transformations, which is based on our knowledge of basic kinetic and mechanistic data and results of suitable environmental chamber experiments is described in a separate document (Reference 1).

The main inputs of concern to the normal user are the conditions of the model scenario (which includes the time schedule for the emissions of the test compound, but not its chemical nature or the total amount emitted), the chemical composition of the test substance, and the amount of test substance emitted.

As indicated in the Introduction, the scenarios for which this model can be applied are limited to the "box model" formulation, illustrated in Figure 2. In this formulation, emissions and chemical reactions are occurring in a single well-mixed air parcel of constant area but variable inversion (or mixing) height which may be undergoing dilution as the inversion height changes, and which may be moving over different areas, as represented by time-varying emissions sources. This is the same type of physical model formulation which is used by the EPA's Empirical Kinetic Modeling Approach (EKMA) models to estimate the amount of total ROG control required to achieve the required reduction in total ozone formation (References 4-6). The EPA supplies a standard set of recommended "default" model inputs to be used with the EKMA models where the appropriate input data are not known. However, it recommends that local agencies use "city-specific" inputs to represent the conditions of the specific airsheds of interest. The problem of deriving the appropriate set of model inputs in this model for defining scenarios for reactivity assessment purposes under the conditions of a specific airshed of interest is entirely analogous, and requires the same types of input data. Indeed, this modeling system could be used to carry out EKMA analyses, though it has not been specifically designed for this purpose.

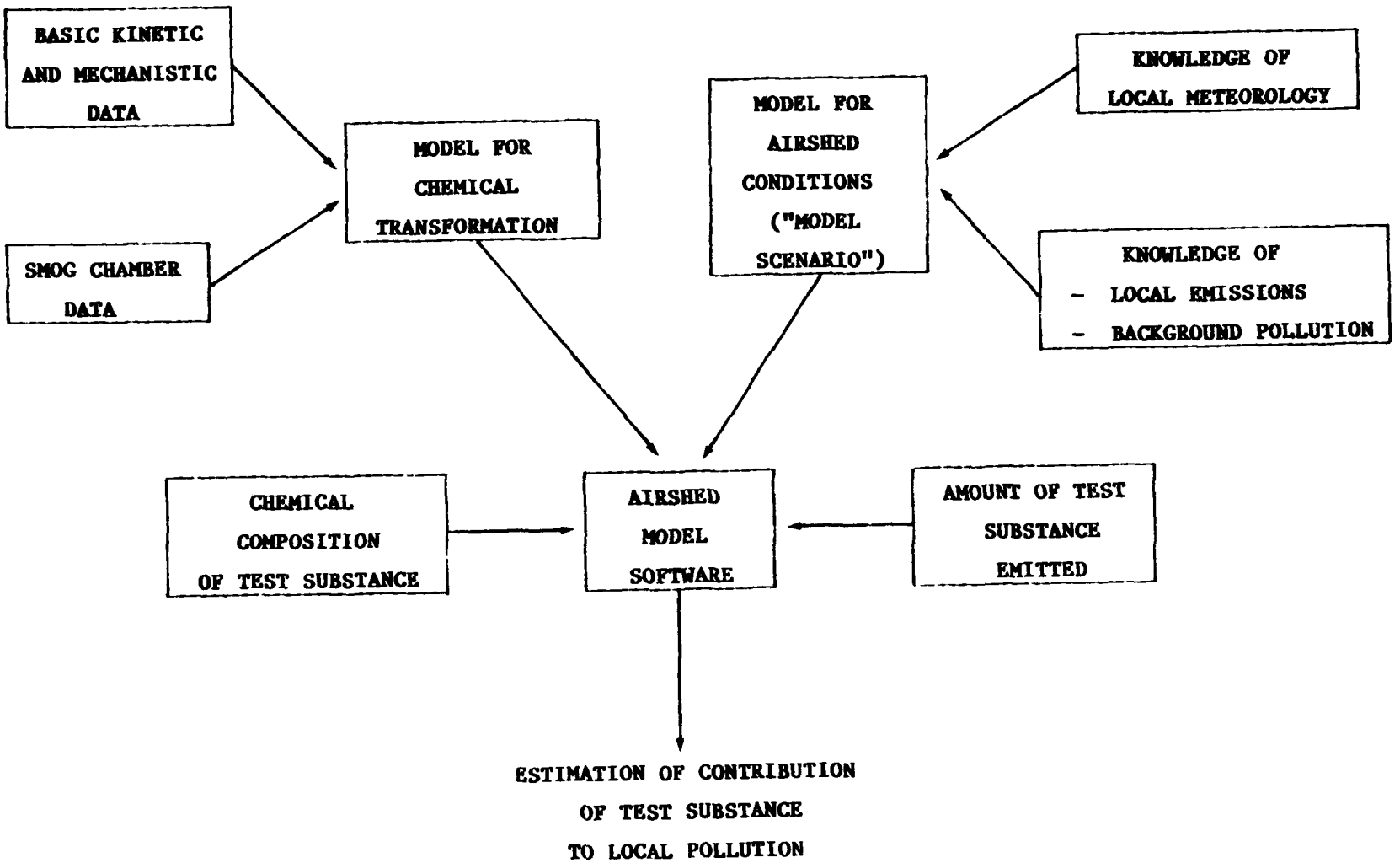


Figure 1. Components of Airshed Model for Estimating Atmospheric Reactivities.

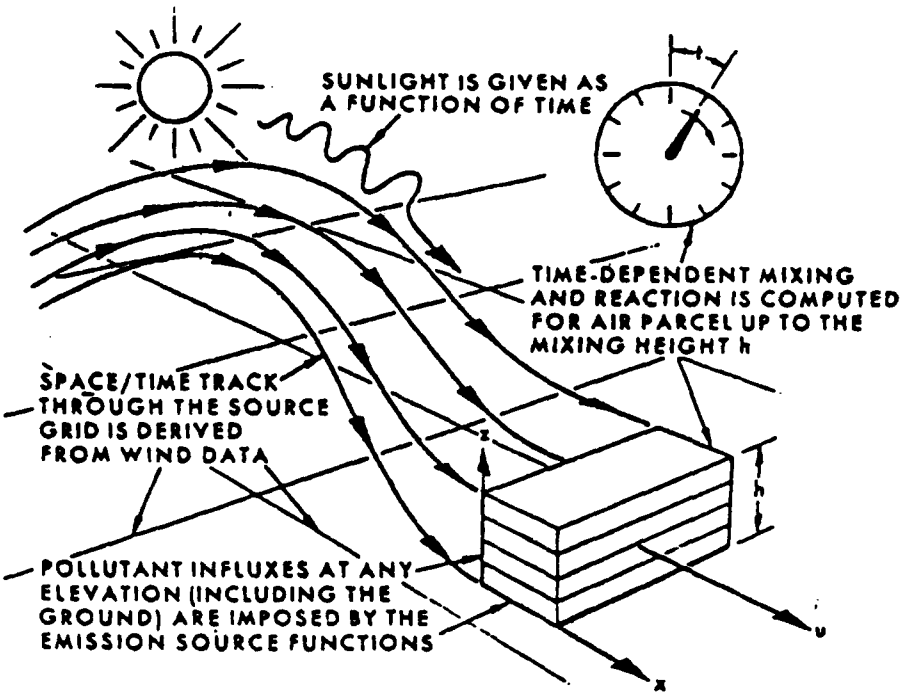


Figure 2. Illustration of the "Floating Box" Model Formulation (Reference 3).

The software in this system allows for one of the parameters of the scenario to have a range of values, to allow for easy assessment of how reactivity changes as the value of that parameter varies. The varied parameter could be the amount of NO_x or base case ROG pollutants emitted, the light intensity factor, and the inversion height (if constant). However, in the example scenarios included with the initial distribution of this model, the total amount of NO_x emitted is among the most important of the parameters affecting the overall reactivities of emitted organics, at least with respect to atmospheric ozone formation (Reference 2). Because of this "varied parameter" option, each scenario, when referenced by this software, can actually be a series of scenarios, each with a different value of a parameter which may significantly affect reactivity. This feature allows a series of reactivity assessment calculations to be carried out as a result of a single command.

The initial distribution of this model includes data files defining several standard scenarios, including three (designated EKMA1 through EKMA3), based on those developed by the EPA to carry out EKMA analyses (Reference 6). These scenarios differ primarily by the amount of dilution assumed to occur, with EKMA1 being the scenario of least dilution, and EKMA3 having the greatest dilution. Also included in this distribution are data files giving the compositions of the six synthetic fuel mixtures and the one synthetic exhaust mixture employed in the environmental chamber experiments used to test this model (References 1, 7). These scenarios and mixtures are described in more detail in Section V, where specific examples of application of this software are given.

Although Section V gives a more complete discussion of examples of application of this model. it is useful to illustrate the type of data which can be obtained with this model. Table 1 gives the maximum daily ozone, PAN, and formaldehyde concentrations calculated in the base-case EKMA1 scenario at the NO_x level most favorable to ozone formation, and shows the percent of change in these yields caused by adding the "standard" synthetic fuel (Reference 7) or the synthetic exhaust mixture (Reference 1) by an amount equal to 10 percent of the base-case ROG emissions in this scenario. Under the conditions of this scenario, this model predicts that the synthetic exhaust mixture is much more reactive to resulting levels of all three of these pollutants than is the synthetic fuel mixture. Additional examples of results of reactivity assessment calculations, showing how reactivities of these and other mixtures vary from scenario to scenario, and as NO_x is varied within a given scenario, are shown in Section V.

The examples discussed above, and in Section V, focus primarily on the applications of this modeling system for reactivity assessment purposes, since the level of detail of the chemical model it employs makes it particularly suitable for this purpose. However, this model can be employed for other applications where box model airshed simulations might be useful, such as (as indicated above) carrying out EKMA analyses, assessing effects of proposed control strategies, assessing atmospheric impacts of proposed new emissions sources, or assessing the contribution of one particular emissions source (such as, for example, an Air Force

TABLE 1. MAXIMUM DAILY OZONE, PAN, AND FORMALDEHYDE CALCULATED FOR THE BASE CASE EKMA1 SCENARIO FOR ROG/NO_X = 8, AND PERCENTAGE CHANGES CAUSED BY ADDITION OF A SYNTHETIC FUEL OR A SYNTHETIC EXHAUST MIXTURE TO THE TOTAL BASE CASE ROG EMISSIONS[a].

Pollutant	Base case maximum (ppb)	Percent change from fuel or exhaust addition	
		Synthetic fuel	Synthetic exhaust
Ozone	204.0	4.5	12.9
PAN	9.7	8.2	31.8
Formaldehyde	25.0	0.0	9.6

[a]Total ROG emissions in base case simulation = 12 millimole C m^{-2} day^{-1}. Amount of synthetic fuel or synthetic exhaust added in the reactivity assessment calculations = 1.2 millimole C m^{-2} day^{-1}.

Base) on overall pollution levels in surrounding areas. Some of these applications require much more detailed information regarding the exact conditions of the airshed of interest and its various pollution sources than that required for reactivity assessment calculations. For some purposes, models with more complex representations of spacial variations, meteorology, and transport conditions (and thus, more simplified representations of the chemical transformations than would be appropriate for reactivity assessment models), might be more appropriate. Users must be aware of the relative strengths and weaknesses of any model in determining whether it is appropriate for their applications.

B. MAJOR COMPONENTS OF THE MODEL SOFTWARE

The USAF atmospheric photochemical reactivity modeling system described in this document incorporates a series of related computer programs and data files whose interrelationships are shown diagrammatically in Figure 3. Some of these programs and data files (e.g., the Model Preparation and Integration Programs, and the batch mode input files) are obvious to the user. However, a general familiarity with the entire system is useful in understanding how the system works, and what can be done with it. The major components of the system are described below.

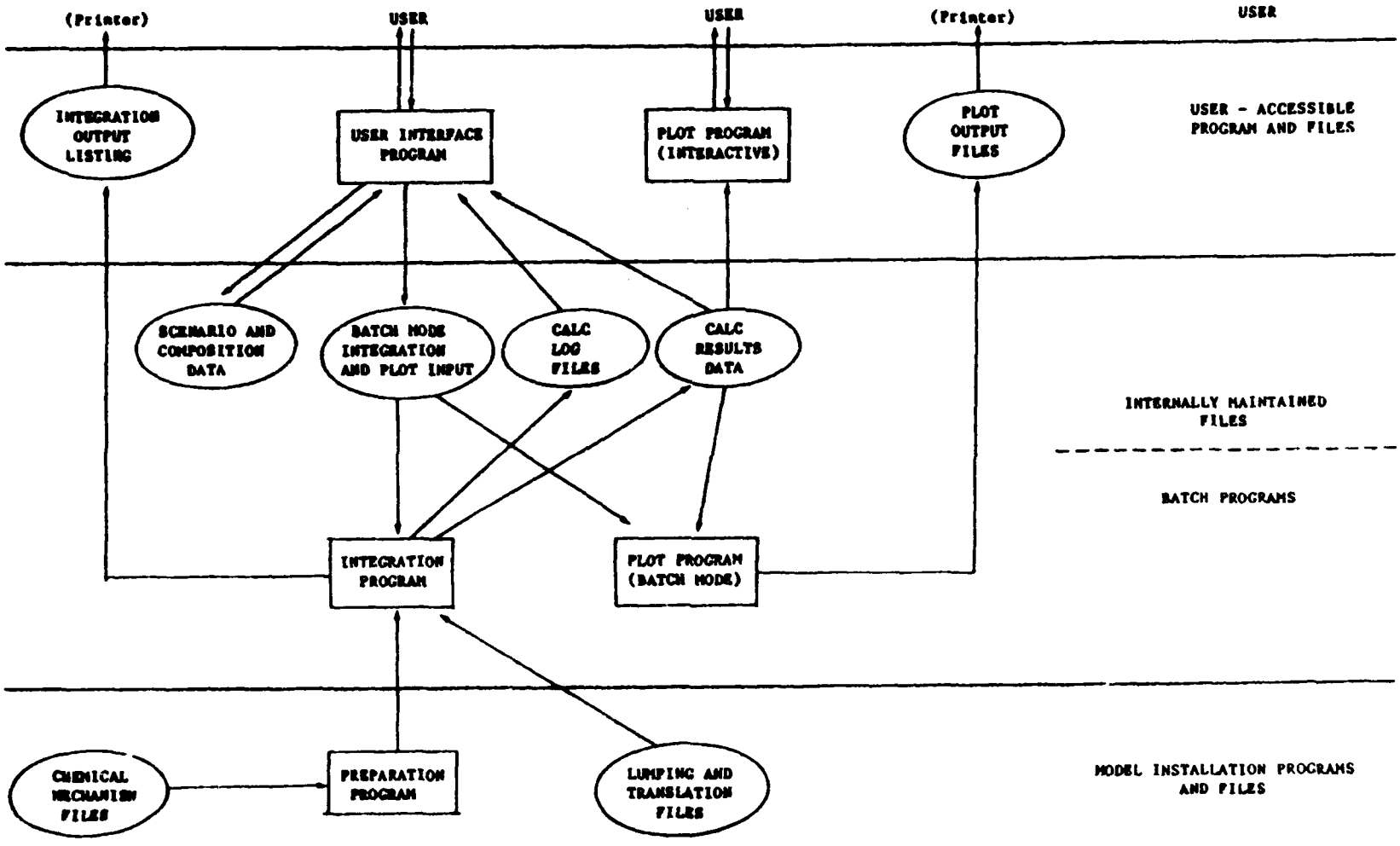


Figure 3. Components of the SAPRC-USAF Atmospheric Photochemical Modeling System and their Interrelationship.

1. User Interface Program

The User Interface is an interactive program which the user can use to set up and conduct a wide variety of airshed model simulations employing "floating box" type scenarios. Minimum training is required beyond an understanding of the principles involved in airshed model calculations, and appropriate values for the required input data. Reactivity assessment calculations also require knowledge of the chemical composition of the substances whose atmospheric reactivities are being assessed. The User Interface program is not strictly necessary to operate this modeling system, but is designed to simplify the use for the applications of interest. In particular, it is sufficiently flexible to be employed in the applications indicated above in the Introduction.

The user can employ the User Interface program for the following operations: (a) to define or modify conditions of model scenarios for airshed calculations, including (for example), amounts of base case ROG and NO_x emitted, schedules for base case ROG, NO_x, and test mixture emissions, schedules for how inversion height changes with time, the date and the latitude to use in calculating the light intensity, the composition of base case ROG emissions, the composition of species present initially (background species), the composition of species present aloft which are entrained into the air mass as the inversion height increases, etc.; (b) to define or modify "compositions files," which are used to specify compositions of the test mixtures whose reactivities are to be assessed, the mixtures used to represent the base case ROG emissions, of the background species, and of the aloft species; (c) to give commands to cause the model simulations employing specified model scenarios and test mixtures to be carried out; and (d) to examine, print, plot or tabulate selected results of these calculations, once they have been completed. The calculations are carried out externally to the program, using the model integration program discussed below.

The User Interface program indicates the type of input it expects at various points, checks for appropriate input, has a "HELP" feature to assist the user when necessary, and can optionally prompt the user for values for each of the various parameters required to define scenarios,

leading the user through the process in a stepwise manner. The program allows named scenarios and compositions of mixtures to be saved for subsequent calculations employing different mixtures, scenarios, or relative emission levels. Calculations employing scenarios and mixtures which have already been defined can be started using single commands. The system is delivered with several representative scenarios and mixtures already defined. The user can employ these to carry out calculations immediately, or use these as starting points to define scenarios or mixtures of interest with respect to particular applications.

2. Plotting Program

The Plotting program can be used to produce page plots of concentration-time data calculated by the model for selected pollutants, and to tabulate selected results of these calculations. It can also be used to plot results of chamber experiments, model simulations of chamber experiments, or the results of one model simulation against another. The features of the plot program are also available as part of the User Interface program. To use the Plotting program, the user must know the calculation number assigned to the particular simulation. The calculation number is given by the User Interface when the user issues the CALC command, and is given in the output of the Integration program. The user can also use the LOG command in the User Interface to examine the calculation logging file which gives the calculation numbers for the various simulations which have been carried out. Alternatively, if the calculation was recently conducted, the user can employ zero or a negative number as a "relative" calculation number, which indicates the calculation number relative to the last completed calculation.

The Plotting program can be used in one of three ways.

(a) On an interactive basis, the user gives the list of compounds to be plotted or tabulated, and the calculation numbers (or chamber runs) whose data are to be plotted. The output will then go to the user's terminal, unless it is specified explicitly to produce the output in a file. This program can be run interactively, either by

invoking the PLOT feature of the User Interface, or by giving the system command to run the Plotting program externally to the User Interface.

(b) The user can use the system editor to create an input file to the Plotting program, and then give the system command to run the program using that file as input. The program then produces an output file whose name is determined by the name of the input file (see Section IV).

(c) In doing the model calculations through the User Interface program, the user can include as Output Options names of pollutants to plot and/or tabulate. This will cause the User Interface to automatically run the Plotting program in batch mode. The Plotting output file so produced will either be automatically sent to the system line printer, or saved as a file (referenced by the calculation number), depending on the output options given to the User Interface.

3. Model Integration Program

The Model Integration program actually conducts the model simulations. It requires input data describing the conditions of the model scenario and the chemical species emitted or initially present. It utilizes these data, and the various data files implementing the chemical mechanism, to carry out the simulation by numerically integrating the differential equations involved. The results of the calculations are tabulated in an output file and are stored on disk for subsequent plotting or tabulation. Each calculation which is carried out (with results saved on disk) is assigned a sequential "calculation number," which is used to reference the results of the calculation when using the User Interface or the Plotting program.

The user can utilize the Integration program in one of two ways: (a) by employing the User Interface program to produce the Integration program's input file and to automatically give the system commands to run the Integration program in batch mode or real time, or (b) by using the system editor to create or modify the Integration program input file specifying the model scenario and emitted species, and then giving the explicit system commands to cause the program to be run. This program is

usually run in a batch mode since, running this program can take significant computer time, depending on the type of computer and the complexity of the calculation.

In either case, the Integration program does not prompt for any input from the user except for the name of the input file (if it was not given on the command line) and it normally does not produce any output to the user's terminal other than a message indicating how many calculations were conducted. Explicitly creating the Integration program input file and running the program allows for the full capability of the Integration program to be employed, but requires a detailed knowledge of the input requirements of this complex program, and is mainly for the advanced user. Using the User Interface to create the integration input file and run the program is designed to be much more suitable for use by the more inexperienced user, and is sufficiently flexible for most routine applications. This method is documented in this User's Manual, rather than the full capabilities of the Integration program.

4. Model Preparation Program

The Model Preparation program is used to process data files describing the chemical mechanism to produce the model-specific subroutines required by the Integration program. It also produces other data files required to conduct model simulations using the chemical mechanism. The Model Preparation program needs to be run only when installing this system on a new computer, or when the chemical mechanism needs to be changed, and will not normally be run by the user of this modeling system. It is not an interactive program; the user must use the system editor to create or modify the input files; its only normal output to the terminal is a simple message indicating that it is finished. A discussion of the use of this program is beyond the scope of this document.

It should be emphasized that making changes to the chemical mechanism may not only require running the Model Preparation program, but may also require changes in other data sets in this system (particularly if species are added or removed from the model). Thus, such changes should only be made by users with a thorough understanding of all

components of this system. In addition, changing the chemical mechanism without knowing the relevant laboratory data upon which the existing mechanism is based, or without testing the revised model against the appropriate data base, may adversely affect the validity of the model's predictions, and should be done with care.

SECTION III

REPRESENTATION OF CHEMICAL SPECIES IN THE MODEL

In this section, the chemical species represented in the current version of the model are listed, and the methods used to represent other species not explicitly represented in the model are summarized. The chemical mechanisms employed and the principals involved in the representation of the chemical species in this model are documented in detail in the report describing the development of this model (Reference 1), but a full understanding of all the details documented in that report is not necessary to use this model. However, the user must know the types of species whose atmospheric reactions can be represented in this model and their nomenclature, and this is discussed in this section. Such information is required to prepare composition files via the User Interface, and to use the plotting program to display their calculated concentrations. For more detailed information concerning the chemical mechanism and how species are represented in this model, the user should consult the report on the development of this model (Reference 1).

The user should be aware, however, that the chemical mechanism and methods used to represent chemical species are not fixed in the software, but are determined by "model installation files" (see Figure 3, in Section II.B), which are normally transparent to the user, since they only need to be modified when the chemical mechanism is changed. The discussion in this section pertains only to the version of the chemical mechanism described in the report on the initial development of this model (Reference 1), which is incorporated in the initially installed version of this model. Thus, this discussion may not be applicable for later updates of this mechanism, and probably will not be applicable if this software is used with mechanisms developed by other researchers.

A. LUMPING TECHNIQUE FOR ROG SPECIES

This model can represent the reactions of a wide variety of ROG species, including alkanes, alkenes, aromatic hydrocarbons, aldehydes, ketones, and several other heteroatom-containing organics. However, the

chemical model does not contain explicit reactions for all of the many types of species it can be used to represent; instead, an approach called "lumping" is employed where a much smaller number of "model species" (i.e., species whose reactions are explicitly included in the mechanism) are used to represent the reactions of the much larger number of individual compounds or classes of compounds which can be specified in the model input defining ROG compositions. Two different lumping approaches can be employed in this model, the "lumped molecule" approach and the "lumped parameter" approach. In the lumped molecule approach, one compound in the model is used to represent the reactions of other compounds which are chemically similar to it. For example, in this model, methyl ethyl ketone (MEK) is used to represent the reactions of all the simple $>C_4$ ketones, 2,3-dimethylnaphthalene is used to represent all di-substituted naphthalenes, etc. In this approach, the kinetic and kinetic mechanistic parameters of the model compound are the same regardless of which species it is being used to represent.

In the lumped parameter approach, one model compound is also used to represent the reactions of one or more other compounds, but, the kinetic and mechanistic parameters of the model compound are determined by the group of compounds it is being used to represent. In this case, compounds are not necessarily grouped by their chemical class, but by which species they react with in the model, and how rapidly they react. For example, despite the significant differences in the details of their chemistry and the products they form, p-xylene and n-tetradecane both react significantly in the atmosphere only with OH radicals, and they react at comparable rates; under this approach they can be lumped together and represented by a single model compound, whose kinetic and mechanistic parameters are determined by the averages (weighed by the relative amounts of each present in the mixture) of those for p-xylene, n-tetradecane, and other compounds which react primarily with OH radicals in the appropriate rate constant range. Because they react to a significant extent only with OH radicals, alkanes and aromatics which react at similar rates can be lumped together under this approach; but alkenes have to be lumped separately because they can also react significantly with ozone and (to a lesser extent) with $O(^3P)$ atoms and NO_3 radicals.

This approach can be potentially much more accurate than the lumped molecule approach, since, under the lumped parameter approach, the kinetic and mechanistic parameters in the model compounds are tailored to be appropriate the specific group of compounds they are being used to represent, while under the lumped molecule approach the parameters for the model compounds are the same, regardless of what they represent. The lumped parameter approach also has the restriction that only compounds with the exact same emissions schedule can be lumped together, because the model cannot account for changes in appropriate kinetic and mechanistic parameters brought about by time. This can be a significant restriction in applying this approach in model simulations of complex airshed scenarios, where a number of separate emissions sources may be represented as introducing pollutants into the air parcel at different times. In the model formulation used in the airshed calculations created by the User Interface, the net effect of this restriction is that species in the base case ROG emissions cannot be lumped with test organic emissions using this approach, since the User Interface allows them to have different emissions schedules. This can be handled by using separate model species to represent base case ROG species than used to represent the reactions of the species in the test mixtures.

The user of this model has a certain degree of choice as to the type and level of detail of lumping approach to be employed. As initially distributed, four different chemical submodels, or lumping models, can be employed in the calculations carried out through the User Interface. These lumping models are designated LM1ARM, LM2ARM, LPARM, and LPBRM and are selected using the MOD option in the SCEN mode of the User Interface (see Section IV.D for a discussion of the SCEN mode commands in the User Interface). All of these "models" incorporate the same basic reaction mechanisms and can be used to represent the same sets of organic species, but they differ in the level of detail employed in representing the organic species (particularly the higher molecular weight species) and in the type of lumping approach employed. Users should base their choice of which lumping model to employ on the type of problem to which it is being applied and the relative importance of maximizing chemical accuracy vs. minimizing computation time. The differences between these models, and the lumping approaches they employ, are indicated below.

The LM1ARM and LM2ARM chemical submodels are both based on the lumped-molecule approach, but differ in the number of model species employed to represent the reactions of the higher-molecular-weight alkanes and aromatics. (The names are derived from "Lumped-Molecule Atmospheric Reactivity Model.") The LM1ARM model employs what we designate as the "LM1" representation, which is the more condensed of the two, while the LM2ARM model employs the more detailed "LM2" representation. The LM1 representation is consistent with the lumping approach in the version of the model we have developed for the EPA for use in developing ozone control strategies (Reference 8), and is similar to the level of detail used in other airshed models. This representation may in many cases be suitable for representing "base-case" ROG mixtures, provided that they do not contain significant levels of $>C_9$ alkanes, naphthalenes, or $>C_5$ alkenes. The more detailed LM2 representation includes additional species to represent the reactions of the naphthalenes and the higher alkanes and alkenes. This is somewhat more appropriate for representing mixtures, such as jet fuels or jet exhaust mixtures, which contain significant quantities of such compounds. However, the LM2 representation is not as accurate for this purpose as the lumped parameter (LP) representation employed in the other two lumping models. The specific representations used for the various classes of ROG species in these lumping techniques are given in the species tabulations in the following section.

The LPBRM and the LPARM models employ what we designate the "LP" (for Lumped Parameter) representation to represent the reactions of some or all of the emitted organics. (The "LP" representation is actually a combination of the two approaches, with the lumped parameter being used for the higher alkanes, alkenes, and aromatics, and the lumped-molecule approach being used for the other species which are lumped, such as higher oxygenates, etc.) The "LP" representation is the most chemically accurate of the species lumping approaches available in this model, and is the approach employed when testing this model's chemical mechanism against the environmental chamber data (Reference 1). However, as indicated above, the lumped-parameter approach can only be used to lump species with the same emissions schedule, which, in terms of this model, means that in reactivity assessment calculations the base-case ROG species and species

in the test mixture cannot be lumped together. In order that this approach may be used for reactivity assessment calculations, the LPARM model includes a separate set of species such that the base case ROG emissions are represented by the LM1 approach, while the test mixture ROG emissions are represented by the LP approach, for maximum accuracy of representation of the mixture whose reactivities are being assessed. As indicated above, the "LM1" approach is considered to be sufficient for representing base case ROG emissions in most cases, and is suitable for reactivity assessment calculations, where the main concern is accuracy in representing the chemistry of the test mixture.

The LPBRM ("Lumped-Parameter Base Reactivity Model") employs the same lumping representation as employed in the LPARM for the reactions of the species in the test mixture, but differs from the LPARM model in that it does not contain separate sets of model species for representing the ROG species in the test mixtures from those used for the base ROG emissions. Thus it cannot be validly applied for scenarios where the test mixture has a different emissions schedule than do the base ROG emissions. Since the User Interface and the Integration program software are not presently capable for checking this situation, this LPBRM model is formulated in such a way that it cannot be employed with test mixtures, i.e., it can only be used with the TML SCEN option (Section IV.D) = 0. (If used with nonzero TML, the User Interface will give no error message, but the calculation will fail when processed by the integration program.) Thus, this model cannot conveniently be used in reactivity assessment calculations. However, it is useful in checking the effects of using the more approximate LM1 or LM2 representations in the base case simulations, and can be applied to problems where use of a single mixture to represent all ROG emissions is sufficient.

As indicated above, the choice of which lumping model to employ depends on the type of ROG species being modeled and the relative importance of chemical accuracy vs. amount of computer time required to carry out the calculations. The "LP" lumping is the most accurate in representing mixtures, such as jet fuels and exhausts, which contain high molecular weight alkanes and aromatics, although in many cases models using "LM2" lumping may not give significantly different results. The

"LM1" lumping is totally unsuitable for representing mixtures such as jet fuels and exhausts, but is probably suitable for representing base-case ROG emissions and compositions of background species, where generally lower molecular weight compounds dominate. "LM2" lumping requires seven more model species than "LM1" to represent ROG species, while "LP" lumping requires only two more model species than "LM1." Thus, in applications where base-case ROG and test organic emissions can be lumped together, use of the more accurate "LPBRM" model is preferred over the "LM2ARM" model because the former has fewer species, and probably would be preferred over use of the "LM1ARM" model, since the "LPBRM" model has only two more species. However, use of the "LPBRM" model does not allow separate specifications of test mixtures and test mixture levels via the User Interface, which are convenient for reactivity assessment calculations. (The "LPBRM" model could still be used for reactivity assessment -- provided the emissions schedules for base case and test organic emissions are the same -- but the user has to combine the base case and test mixtures into a single composition file which is designated as the BOC SCEN parameter in the User Interface input.) For more convenient use of the "LP" lumping in reactivity assessment calculations, or for cases where separate test mixture emissions schedules are required, the "LPARM" model, which has separate species for base case ROG emissions, should be used. However, the "LPARM" model has two more species than the "LM2ARM" model, so use of the latter may be preferred if it is shown to give the same results for the type of test mixture whose reactivity is being assessed. The "LM2ARM" model is also much more accurate than the "LPARM" model in representing base case ROG species, since the latter uses the more approximate "LM1" lumping for those species.

B. LIST OF ROG AND MODEL SPECIES USED

Table 2 lists all of the different ROG compounds or classes of compounds whose reactions can be represented in the current version of this model, and gives the model species used to represent them under the various lumping techniques as currently implemented. The names given in the first column of this table are the only legal names which can be used

TABLE 2. LIST OF ROG SPECIES OR CLASSES WHICH CAN BE INCLUDED IN COMPOSITION FILES SPECIFYING BASE CASE OR TEST MIXTURE EMISSIONS, AND MODEL SPECIES USED TO REPRESENT THEM.

ROG name[a]	Model species[b] LM1	LM2	LP	Compound or compounds represented
	Unreactive Compounds			
INERT	INERT	INERT	INERT	Unreactive compounds
	Oxygenates and Heteroatom-Containing Organics			
FORMALD	HCHO	HCHO	HCHO	Formaldehyde
ACETALD	CCHO	CCHO	CCHO	Acetaldehyde
PROPALD	RCHO	RCHO	RCHO	Propionaldehyde and other $>C_3$ monoaldehydes
ACROLEIN	RCHO	RCHO	RCHO	Acrolein
ACETONE	C-CO-C	C-CO-C	C-CO-C	Acetone
MEK	MEK	MEK	MEK	Methyl ethyl ketone and all $>C_4$ ketones except alpha-dicarbonyls
GLYOXAL	HCOCHO	HCOCHO	HCOCHO	Glyoxal
MEGLYOX	CCOCHO	CCOCHO	CCOCHO	Methyl glyoxal and other alpha dicarbonyls
PHENOL	PHENOL	PHENOL	PHENOL	Phenol
CRESOL	CRES	CRES	CRES	Cresols and other alkylphenols
BENZALD	BZ-CHO	BZ-CHO	BZ-CHO	Benzaldehyde and other aromatic aldehydes
NITROPHEN	NITROPHEN	NITROPHEN	NITROPHEN	Nitrophenols, nitrocresols, and other aromatic nitro-compounds
FURAN	FURAN	FURAN	FURAN	Furan and related compounds
THIOPHEN	THIOPHEN	THIOPHEN	THIOPHEN	Thiophene and related compounds
PYRROLE	PYRROLE	PYRROLE	PYRROLE	Pyrrole and related compounds
PROPANE	INERT	INERT	BZC3	Propane
N-C4	C4C5-ALK	C4C5-ALK	C4C5	n-Butane
ISO-C4	C4C5-ALK	C4C5-ALK	C4C5	Isobutane
N-C5	C4C5-ALK	C4C5-ALK	C4C5	n-Pentane
ISO-C5	C4C5-ALK	C4C5-ALK	C4C5	Isopentane
NEO-C5	INERT	INERT	BZC3	Neopentane
N-C6	C4C5-ALK	C4C5-ALK	AAR1	n-Hexane
2-ME-C5	C6P-ALK	C6P-ALK	AAR1	2-Methyl pentane
3-ME-C5	C6P-ALK	C6P-ALK	AAR1	3-Methyl pentane
23-DMB	C6P-ALK	C6P-ALK	AAR1	2,3-Dimethyl butane
BR-C6	C6P-ALK	C6P-ALK	AAR1	Unspeciated branched hexanes
CYC-C6	C6P-ALK	C6P-ALK	AAR2	C_6 cycloalkanes
N-C7	C6P-ALK	C6P-ALK	AAR1	n-Heptane
3-ME-C6	C6P-ALK	C6P-ALK	AAR1	3-Methyl hexane
4-ME-C6	C6P-ALK	C6P-ALK	AAR1	4-Methyl hexane
24-DM-C5	C6P-ALK	C6P-ALK	AAR1	2,4-Dimethyl pentane

TABLE 2. LIST OF ROG SPECIES OR CLASSES WHICH CAN BE INCLUDED IN COMPOSITION FILES SPECIFYING BASE CASE OR TEST MIXTURE EMISSIONS, AND MODEL SPECIES USED TO REPRESENT THEM (CONTINUED).

ROG name[a]	Model species[b] LM1	LM2	LP	Compound or compounds represented
23-DM-C5	C6P-ALK	C6P-ALK	AAR1	3,4-Dimethyl pentane
ME-CYCC6	C6P-ALK	C6P-ALK	AAR2	Methyl cyclohexane
BR-C7	C6P-ALK	C6P-ALK	AAR1	Unspeciated branched heptanes
CYC-C7	C6P-ALK	C6P-ALK	AAR2	Unspeciated C_7 cycloalkanes
N-C8	C6P-ALK	C6P-ALK	AAR2	n-Octane
4-ME-C7	C6P-ALK	C6P-ALK	AAR2	4-Methyl heptane
ISO-C8	C6P-ALK	C6P-ALK	AAR1	2,2,4-Trimethyl pentane
ET-CYCC6	C6P-ALK	C6P-ALK	AAR2	Ethyl cyclohexane
BR-C8	C6P-ALK	C6P-ALK	AAR2	Unspeciated branched octanes
CYC-C8	C6P-ALK	C6P-ALK	AAR2	Unspeciated C_8 cycloalkanes
N-C9	C6P-ALK	C9P-ALK	AAR2	n-Nonane
4-ET-C7	C6P-ALK	C9P-ALK	AAR2	4-Ethyl heptane
BR-C9	C6P-ALK	C9P-ALK	AAR2	Unspeciated branched nonanes
CYC-C9	C6P-ALK	C9P-ALK	AAR3	Unspeciated C_9 cycloalkanes
N-C10	C6P-ALK	C9P-ALK	AAR3	n-Decane
4-PR-C7	C6P-ALK	C9P-ALK	AAR2	4-Propyl heptane
BR-C10	C6P-ALK	C9P-ALK	AAR2	Unspeciated branched decanes
CYC-C10	C6P-ALK	C9P-ALK	AAR3	Unspeciated C_{10} cycloalkanes
N-C11	C6P-ALK	C9P-ALK	AAR3	n-Undecane
BR-C11	C6P-ALK	C9P-ALK	AAR3	Unspeciated branched undecanes
CYC-C11	C6P-ALK	C9P-ALK	AAR3	Unspeciated C_{11} cycloalkanes
N-C12	C6P-ALK	C9P-ALK	AAR3	n-Dodecane
BR-C12	C6P-ALK	C9P-ALK	AAR3	Unspeciated branched docecanes
CYC-C12	C6P-ALK	C9P-ALK	AAR3	Unspeciated C_{12} cycloalkanes
N-C13	C6P-ALK	C9P-ALK	AAR4	n-Tridecane
BR-C13	C6P-ALK	C9P-ALK	AAR3	Unspeciated branched tridecanes
CYC-C13	C6P-ALK	C9P-ALK	AAR4	Unspeciated C_{13} cycloalkanes
N-C14	C6P-ALK	C9P-ALK	AAR4	n-Tetradecane
BR-C14	C6P-ALK	C9P-ALK	AAR3	Unspeciated branched tetradecanes
CYC-C14	C6P-ALK	C9P-ALK	AAR4	Unspeciated C_{14} cycloalkanes
N-C15	C6P-ALK	C9P-ALK	AAR4	n-Pentadecane
BR-C15	C6P-ALK	C9P-ALK	AAR3	Unspeciated branched pentadecanes
CYC-C15	C6P-ALK	C9P-ALK	AAR4	Unspeciated $>C_{15}$ cycloalkanes
BENZENE	BENZ	BENZ	BZC3	Benzene
TOLUENE	TOLU	TOLU	AAR1	Toluene
M-XYLENE	M-XYEN	M-XYEN	AAR3	m-Xylene
O-XYLENE	M-XYEN	M-XYEN	AAR2	o-Xylene
P-XYLENE	M-XYEN	M-XYEN	AAR2	p-Xylene
123-TMB	M-XYEN	135-TMB	AAR5	1,2,3-Trimethylbenzene
124-TMB	M-XYEN	135-TMB	AAR5	1,2,4-Trimethylbenzene
135-TMB	M-XYEN	135-TMB	AAR5	1,3,5-Trimethylbenzene
ALK1BENZ	TOLU	TOLU	AAR1	Monoalkyl benzenes

TABLE 2. LIST OF ROG SPECIES OR CLASSES WHICH CAN BE INCLUDED IN COMPOSITION FILES SPECIFYING BASE CASE OR TEST MIXTURE EMISSIONS, AND MODEL SPECIES USED TO REPRESENT THEM (CONCLUDED).

ROG name[a]	Model species[b] LM1	LM2	LP	Compound or compounds represented
ALK2BENZ	M-XYEN	135-TMB	AAR3	Dialkyl benzenes
ALK3BENZ	M-XYEN	135-TMB	AAR5	Tri- and poly-alkyl benzenes
TETRALIN	M-XYEN	TETRALIN	AAR3	Tetralin. Also used for indans
NAPHTHAL	M-XYEN	NAPH	AAR4	Naphthalene
ME-NAPH	M-XYEN	ME-NAPH	AAR5	Monoalkyl naphthalenes
DM-NAPH	M-XYEN	23-DMN	AAR5	Di- and poly-alkyl naphthalenes
ETHENE	C:C	C:C	C:C	Ethene
PROPENE	C:CC	C:CC	OLE1	Propene
1-BUTENE	C:CC	C:CC	OLE1	1-Butene
C-2-BUTE	CC:CC	CC:CC	OLE2	cis-2-Butene
T-2-BUTE	CC:CC	CC:CC	OLE2	trans-2-Butene
ISOBUTEN	C:CC	C:CC	OLE1	Isobutene
1-PENTEN	C:CC	C:CC	OLE1	1-Pentene
2M-1-BUT	C:CC	C:CC	OLE1	2-Methyl-1-butene
2M-2-BUT	CC:CC	CC:CC	OLE2	2-Methyl-2-butene
1-HEXENE	C:CC	1-HEXEN	OLE1	1-Hexene
23M2-BUT	CC:CC	1-HEXEN	OLE2	2,3-Dimethyl-2-butene
C6-OLE1	C:CC	1-HEXEN	OLE1	Unspeciated C_6 1-alkenes
C6-OLE2	CC:CC	1-HEXEN	OLE2	Unspeciated C_6 2-alkenes
C7-OLE1	CC:C	1-HEXEN	OLE1	Unspeciated C_7 1-alkenes
C7-OLE2	CC:CC	1-HEXEN	OLE2	Unspeciated C_7 2-alkenes
C8-OLE1	CC:C	1-HEXEN	OLE1	Unspeciated C_8 1-alkenes
C8-OLE2	CC:CC	1-HEXEN	OLE2	Unspeciated C_8 2-alkenes
C9-OLE1	CC:C	1-HEXEN	OLE1	Unspeciated C_9 1-alkenes
C9-OLE2	CC:CC	1-HEXEN	OLE2	Unspeciated C_9 2-alkenes
C10-OLE1	CC:C	1-HEXEN	OLE1	Unspeciated C_{10} 1-alkenes
C10-OLE2	CC:CC	1-HEXEN	OLE2	Unspeciated C_{10} 2-alkenes
C11-OLE1	CC:C	1-HEXEN	OLE1	Unspeciated C_{11} 1-alkenes
C11-OLE2	CC:CC	1-HEXEN	OLE2	Unspeciated C_{11} 2-alkenes
C12-OLE1	CC:C	1-HEXEN	OLE1	Unspeciated C_{12} 1-alkenes
C12-OLE2	CC:CC	1-HEXEN	OLE2	Unspeciated C_{12} 2-alkenes
C13-OLE1	CC:C	1-HEXEN	OLE1	Unspeciated C_{13} 1-alkenes
C13-OLE2	CC:CC	1-HEXEN	OLE2	Unspeciated C_{13} 2-alkenes
C14-OLE1	CC:C	1-HEXEN	OLE1	Unspeciated C_{14} 1-alkenes
C14-OLE2	CC:CC	1-HEXEN	OLE2	Unspeciated C_{14} 2-alkenes
C15-OLE1	CC:C	1-HEXEN	OLE1	Unspeciated C_{15} 1-alkenes
C15-OLE2	CC:CC	1-HEXEN	OLE2	Unspeciated C_{15} 2-alkenes

[a]Name which must be used when specifying the species or class of compounds in composition files processed by the User Interface.

[b]Name of species used in the model to represent this ROG species or class, for the LM1, LM2, or LP lumping approaches. If only one name is given, then the same model species is used in all models.

to designate species or classes of compounds in composition files specifying ROG emissions (i.e., in composition files referenced by the BOC or TMC scenario parameters, discussed in Section IV.D.5). These names should also be used to specify levels of these compounds or classes of compounds in composition files giving background or aloft pollutant levels (referenced by the BMC and ALO scenario parameters), although can also include specifications of concentrations of inorganic or nonemitted product organic species which cannot be included in the composition files specifying ROG emissions. The inorganic and nonemitted product organic species which can be included in composition files giving background or aloft pollutant levels are listed in Table 3.

No other species names other than those listed on the first columns of Tables 2 and 3 can be entered in composition files via the user interface. (The user can obtain a list of these species names while in the COMP mode of the User Interface by entering the LIST command, and he can obtain a display of these names along with a description of the compounds they represent by entering the ANNOT command, as discussed in Section IV.E). If the user gives a name not on this list, the program gives an error message, and the name is not entered. However, the User Interface has no way of determining whether a composition file is to be used to specify emissions (where only the names in Table 2 can be used) or to specify background or aloft levels (where the names in Table 3 can be used as well) and thus, will give no error message if the user inappropriately enters a name from Table 2 in a file intended to be used to represent ROG emissions. On the other hand, the User Interface will give an error message if an attempt is made to NORMalize a composition file containing inorganic or nonemitted organic product species, since that command is only appropriate for files intended to specify ROG emissions. Calculation will be rejected by the Integration program if inappropriate species are included in composition files.

The user must recognize the difference between ROG species names which can be entered in composition files and the names of the "model species" used by the Integration program to represent them. In most cases, different names are used for ROG species which can be included in composition files then used for model species which represent them in the

TABLE 3. LIST OF MODEL COMPOUNDS USED TO REPRESENT INORGANIC AND NONEMITTED ORGANIC SPECIES IN THE CHEMICAL TRANSFORMATION MODELS.

Model name	Compound or compounds represented
	Inorganic Species
O3	Ozone
NO	Nitric Oxide[a]
NO2	Nitrogen Dioxide[a]
HONO	Nitrous Acid
HONO2	Nitric Acid
N2O5	Nitrogen Pentoxide
NO3	Nitrate Radicals
HO2NO2	Peroxy Nitric Acid
HO2H	Hydrogen Peroxide
HO.	Hydroxyl Radicals[b]
HO2.	Hydroperoxyl Radicals[b]
	Non-Emitted Organic Products and Intermediates
C-PAN	Peroxyacetyl Nitrate
C2-PAN	Peroxypropionyl Nitrate and other PAN analogues
BZ-PAN	Peroxy Benzoyl Nitrate
HCO-PAN	PAN analogue formed from glyoxal
RONO2	$C4^+$ Alkyl Nitrates
NITROPHEN	Nitrophenols, nitrocresols, and other aromatic nitrocompounds
-OOH	Total organic hydroperoxides[b]
RO2.	Total organic peroxy radicals[b]
RCO3.	Total organic acyl peroxy radicals[b]
HCOCHO(X)	Pseudo-specie used to represent uncharacterized ring-opened products not containing methyl groups[b]
CCOCHO(X)	Pseudo-specie used to represent uncharacterized ring-opened products with methyl groups[b]
HET-UNKN	Pseudo-specie used to represent uncharacterized photoreactive products in the reactions of furan, thiophene, and pyrrole[b]
HET-UNK2	Pseudo-specie used to represent uncharacterized products in the reactions of thiophene and pyrrole which react rapidly with NO^3 radicals[b]

[a] NO and NO_2 emissions are specified through the NO_X and the NO_2 SCEN parameters, and not through composition files.
[b] Cannot be used in composition files.

Integration program. The user does not need to know the names used in the model to represent the ROG species when creating composition files through the User Interface, since the Integration program translates them into the names (and, for models using "LP" lumping, parameter values) it recognizes when carrying out the calculations. However, the results of the calculations of ROG species concentrations are saved only in terms of the model species, since these are the only species for which concentration-time data are calculated. Thus, when examining the results of the calculations to determine ROG concentration-time data, such as when using the plotting program or the PLOT commands in the User Interface, the user must know how the ROG species are lumped, and the names given to the model species used to represent them.

The names of the model species used to represent the various ROG species under the three lumping techniques implemented in this model are given with the ROG species list in Table 2. In addition, the ROG model names used by the three lumping techniques are listed separately in Table 4, which also gives the general types of compounds they are used to represent. These names, along with the names of the inorganic and non-emitted organic product species listed in Table 3 are the only names which will be recognized by the plotting program when using this model. Note that for most organics the model names may be different depending on the lumping, and the user must use the names appropriate for the lumping technique employed when obtaining plots of tabulations of concentration-time data for such species via the Plotting program.

TABLE 4. LIST OF MODEL COMPOUNDS USED TO REPRESENT EMITTED ROG SPECIES IN THE CHEMICAL TRANSFORMATION MODELS.

Model name	Compound or compounds represented
	Emitted Organics Included in All Models
CO	Carbon Monoxide
C:C	Ethene
HCHO	Formaldehyde
CCHO	Acetaldehyde
RCHO	Propionaldehyde and other $>C_3$ aldehydes, including acrolein, but not 1,2-dicarbonyls
BZ-CHO	Benzaldehyde and other aromatic aldehydes
C-CO-C	Acetone
MEK	Methyl ethyl ketone and all $>C_4$ ketones except alpha-dicarbonyls
HCOCHO	Glyoxal
CCOCHO	Methyl glyoxal and other alpha-dicarbonyls
PHENOL	Phenol
CRES	Cresols and other alkylphenols
FURAN	Furan and compounds which can be represented by furan
THIOPHEN	Thiophene and related compounds
PYRROLE	Pyrrole and related compounds
	Species Used to Represent Other Emitted Organics in Models with LM1 Lumping (LM1ARM model and base case ROG species in the LPARM model)
C4C5-ALK	C_4 - C_5 Alkanes
C6P-ALK	$>C_6$ Alkanes
C:CC	Propene and all other 1-alkenes
CC:CC	trans-2-Butene and all internal alkenes
BENZ	Benzene
TOLU	Toluene and other monoalkyl benzenes
M-XYEN	m-Xylene and other di- and poly-alkyl benzenes, and naphthalenes
	Species Used to Represent Other Emitted Organics in Models with LM2 Lumping (LM2ARM model)
C4C5-ALK	C_4 - C_5 Alkanes
C6P-ALK	C_6 - C_8 Alkanes
C9P-ALK	$>C_9$ Alkanes

TABLE 4. LIST OF MODEL COMPOUNDS USED TO REPRESENT EMITTED ROG SPECIES IN THE CHEMICAL TRANSFORMATION MODELS (CONCLUDED).

Model name	Compound or compounds represented
C:CC	Propene and C_3 - C_5 1-alkenes
CC:CC	trans-2-Butene and all other C_4 - C_5 internal alkenes
1-HEXEN	1-Hexene and other $>C_6$ alkenes
BENZ	Benzene
TOLU	Toluene and other monoalkyl benzenes
M-XYEN	m-Xylene and other dialkyl benzenes
135-TMB	1,3,5-Trimethylbenzene and other tri- and poly-alkyl benzenes
NAPH	Naphthalene
ME-NAPH	Monoalkylnaphthalenes
23-DMN	2,3-Dimethylnaphthalene and other di-alkyl naphthalenes
TETRALIN	Tetralin
	Lumped Parameter Pseudo-Species Used to Represent Other Emitted Organics in Models with LP Lumping (LPBRM model and test organics in the LPARM model)
BZC3	Alkanes and aromatics which react with k_{OH} less than 2×10^{-12} cm^3 $molecule^{-1}$ sec^{-1}
C4C5	Alkanes and aromatics which react with k_{OH} between 2.0 and 5.0×10^{-12} cm^3 $molecule^{-1}$ sec^{-1}
AAR1	Alkanes and aromatics which react with k_{OH} between 5.0 and 7.5×10^{-12} cm^3 $molecule^{-1}$ sec^{-1}
AAR2	Alkanes and aromatics which react with k_{OH} between 7.5 and 12.0×10^{-12} cm^3 $molecule^{-1}$ sec^{-1}
AAR3	Alkanes and aromatics which react with k_{OH} between 1.2 and 2.0×10^{-11} cm^3 $molecule^{-1}$ sec^{-1}
AAR4	Alkanes and aromatics which react with k_{OH} between 2.0 and 3.0×10^{-11} cm^3 $molecule^{-1}$ sec^{-1}
AAR5	Alkanes and aromatics which react with k_{OH} greater than 3.0×10^{-11} cm^3 $molecule^{-1}$ sec^{-1}
OLE1	1-Alkenes
OLE2	Alkenes with internal double bonds

SECTION IV

USER INTERFACE PROGRAM

A. OVERVIEW AND SUMMARY OF TERMS AND CONCEPTS

The User Interface program represents a convenient method for the user to employ this model to carry out reactivity assessment calculations. In particular, this program can be employed to:(a) define or modify the conditions of model scenarios for use in airshed calculations, particularly for the purpose of reactivity assessment, (b) specify the compositions of mixtures of emitted pollutants, (c) conduct model simulations employing the model scenarios and defined reaction mixtures, and (d) examine selected results of these calculations. Before discussing the operation of this program, it is useful to summarize the major terms and concepts which must be understood in order to be able to successfully utilize this User's Interface. These terms and concepts are referred to throughout the remainder of this manual.

1. MAIN, SCEN, COMP, and PLOT Modes

The User Interface program has three main modes of operation, each of which accepts different types of input from the user, and each of which is used for a different purpose. In addition, a number of commands prompt the user to answer specific questions which had been initiated by commands previously given from one of the main modes. These commands are discussed in the various sections describing the detailed operations which can be carried out in each mode.

The mode which is entered when the program is first invoked is referred to as the MAIN mode. Most commands other than those specifically involved in creating and editing scenarios or composition files can be given from MAIN mode. Commands that can be given from the MAIN mode include various utility commands such as listing calculations or examining output files, telling the system whether a printer is being used, and examining the status of calculations. Other examples include the CALC command to conduct the calculation, commands to enter the SCEN, COMP or PLOT modes, and commands to exit or abort the program.

The SCEN mode is used to define the parameters for a new scenario, or to modify the parameters in an existing one. The COMP mode is used to define the composition files defining mixtures of compounds which can be referenced in the scenarios. The PLOT mode is used to obtain plots or tabulations of the results of the calculations, given the model species names (see previous section) and the calculation number. The concepts of "scenario," "composition file," and "calculation" when used in the context of this program are discussed below.

2. Scenario

The term "scenario" is used to refer to the total set of conditions or parameters which must be specified to the Integration program in order for it to conduct model simulation. However, in the context of reactivity assessment calculations, the term "scenario" is also used to refer to the set of conditions or parameters other than the composition of the test mixture or the amount of test mixture added (see below). Parameters which make up a scenario include such physical factors as temperature and inversion height, chemical parameters such as total emission levels, and names of composition files specifying which pollutants are emitted. Some of these parameters can vary with time, i.e., have "schedules." One of these parameters, called the "varied parameter," can take on several values so that a series of calculations can be carried out as part of that scenario. Scenarios are defined and their parameters can be modified or edited by entering the SCEN mode of the User Interface program. Scenarios can be stored for future use, and several representative scenarios are included with the modeling system.

3. Composition Files

Composition Files contain a list of names of chemical species recognized by the model and their relative composition or absolute concentrations or emission rates. These files are referenced by name as part of the scenarios, and are created or edited by using the COMP mode of the User Interface. Test mixture ID composition files can optionally be

referenced in the CALC command. Composition files are referenced as part of the scenario to specify which ROG species are emitted, and which pollutants are present initially or in the aloft air mass.

There are essentially two types of composition files: those which represent emissions, and those which specify background and aloft species pollutant concentrations. The emission composition files are used to specify either (a) the composition base case ROG emissions, i.e., either all organic emissions, or, in reactivity assessment calculations, all organic emissions other than emissions of the test organic mixture; or (b) the composition of "test" organic mixtures whose effects on air quality, when added on top of base case emissions, are being calculated in reactivity assessment calculations. Regardless of whether they are being used for base case or test ROG emissions, composition files specifying emissions should give the relative concentrations of species in the mixture, and for proper use of this program, these files should be "normalized," i.e., the number of carbons in the mixture should sum to 1. Normalization of a composition file to be used to specify emissions can be easily accomplished by the user by using the NORM command when editing or creating the file in the COMP mode. However, this is not done automatically, and the program does not check to determine if composition files specifying emissions are properly normalized when the calculation is submitted via the CALC command, or carried out by the Integration program. Composition files specifying ROG emissions can only be names of ROG species which are recognized by the model. The legal ROG species names recognized by the initially distributed version of the model are listed in Table 2, above.

Composition files specifying background or aloft species, unlike those specifying emissions, should contain absolute pollutant concentrations, and they must be in ppm concentration units. They should not be normalized. In addition, also unlike emissions composition files, they can (and in general will) contain concentrations of inorganic and non-emitted organic species (i.e., those listed in Table 3, above, for the current version of the model), as well as the ROG species such as those listed in Table 2. Note that in this context, "background" species refer to pollutants which are present in the air mass before any emissions

occur, and "aloft" species refer to species assumed to be initially present in the air mass above the inversion height, which are entrained into the air mass being simulated as the inversion height rises throughout the day. (The composition of species in the aloft air mass has to be assumed to be constant in this model formulation.) Separate composition files can be used to specify levels of background or aloft pollutants.

4. Calculations and the CALC Command

Once the scenario and all the composition files have been created, the user can give the CALC command to cause model calculations to be carried out. Strictly speaking, the CALC command does not start the calculation, but it causes the User Interface program to create the Integration program input files using named scenarios, and to create the commands which direct the model simulation to be carried out by the Integration program, normally in batch mode. Results from such batch calculations become available to the user after the calculation is completed. The model calculation itself is carried out externally to the User Interface program, but in the subsequent discussion we will consider invoking CALC command as equivalent to actually doing the model simulation. When giving this command, the user must give the scenario name, and can optionally give the composition file and total emission level of a test mixture whose reactivity is being assessed, and information to include on the calculation label (see below) to aid in documenting this calculation.

5. Varied Parameter

The Varied Parameter is a parameter in a scenario which can take on several values. A scenario can have a maximum of one varied parameter, and the parameters which can be varied include the light factor (LF), the inversion height (IH), the humidity (RH), the composition file name used for aloft species (ALO), or the total NO_x levels (NOX). It is not necessary that a scenario have any varied parameters. If a scenario has a varied parameter, each time a CALC command is given for this scenario, a

series of calculations are conducted, one for each of the values given for the varied parameter. In the example scenarios given with this program, the varied parameter is the total NO_x level. Thus, each CALC command causes calculations to be carried out for a series of ROG/NO_x ratios. [This is useful because the reactivity of organics can change significantly depending on the total NO_x level (Reference 2).] The varied parameter is set by giving a parameter name (LF, IH, RH, ALO, or NOX) as value to the VAR parameter in SCEN mode.

6. Schedules

Several parameters in a scenario can vary with time, and the way in which their values vary with time is referred to as their "schedule." Parameters which can optionally be scheduled are the inversion height (IH), light factor (LF), temperature (TEM), NO_x emissions (NXS), base case emissions (BOS), or test organic emissions (TMS). Schedules consist of arrays of ordered pairs, where the first value is the clock time, and the second value is the parameter value at that time. The user can specify whether the values of the parameters used for times between those given are to be determined in a linear manner, with the value being determined by linear interpolation between the values specified in the schedule, or in a stepped manner, where the value given in the schedule is held constant until the time a new value is given in the schedule. Schedules for emissions also include a parameter specifying what fraction of the emitted species are present initially, and what fraction are emitted throughout the day. Schedules must be given for emissions unless all the "emitted" pollutants are to be represented as being present initially.

7. Calculation Number

The Calculation Number is a number assigned to each calculation (or series of calculations) carried out using the CALC command. The user must use these calculation numbers to reference the calculation in order to examine its results. The program informs the user of the calculation number after a CALC command is successfully executed (i.e., after the

files required to do a calculation are created). The user can also give the LOG command to examine a log of calculations which contain the calculation numbers, the associated scenario name, and parameters given in the CALC command.

The calculation number is a positive integer whose value can range from 1 and 9999. Each time a CALC command is given, the calculation number is incremented by the number of simulations which will be carried out. The number of simulations conducted for a given CALC command is either one, if there is no "varied parameter" in the scenario, or the number of values specified for the varied parameter. If there is a varied parameter, each calculation in the set has its own calculation number, but the whole set is referenced by the number of the last calculation in the set. If the calculation number is getting close to 9999, it can be reset by editing the files SUBMIT.PRM and CALC.PRM on DEC systems or SBMTPRM and CALCPRM on the CYBER. (SUBMIT.PRM or SBMPRM contains the last calculation number created through the User Interface, while CALC.PRM or CALCPRM contains the last calculation processed by the integration program.)

A Relative calculation number can also be used to reference calculation results. A calculation number is "relative" if it is zero or negative. A relative calculation number of zero refers to the last calculation which has been completed by the integration program, -1 refers to the second-to-last, etc. Note that if calculations are pending or failed (see below), the last calculation completed by the integration program, which is used in calculating the absolute calculation number from the relative value input, will not be the same as the last calculation submitted by the CALC command.

Note that if a calculation fails, the calculation numbers updated by the User Interface program are not reset, and the next time the user gives CALC commands, the numbers assigned to the new calculations would be the same as if the previous calculations had succeeded. Thus, some calculation numbers may refer to failed calculations.

8. Calculation Label

The Calculation Label is a character string giving the scenario name and the names and values of all parameters specified on the CALC command which initiated the calculation. Additional information that the user wants to include on the calculation label can also be given as part of the CALC command. The calculation label is stored with the results of the calculation, and is printed (along with the calculation number) on the Plotting program output in order to identify the calculation. Ideally, it should be unique for each calculation to be more meaningful to the user than just the calculation number.

9. Status of Submitted Calculations

A calculation which is created by the CALC command in the User Interface can be either Pending, Successful or Failed, depending on whether it has been successfully processed by the integration program. A Pending Calculation is a calculation which has been prepared for execution by the CALC command, but which has not yet been run by the integration program. It is not submitted for execution until the user leaves the user interface program with the EXIT command, and, depending on how busy the computer is and the amount of time needed to carry out the calculation, some time may elapse between the submission of this calculation and its completion. A Successful Calculation is a calculation which has been executed by the Integration program, and whose results have been stored. The user can determine if a calculation is successful either by using the CALCNO command to see the last calculation executed, or by using the LOG command to see if its label has been placed in the calculation log file. The user can then examine the results of the calculation, either by listing or printing the Integration program (or Plotting program) output files, or by running the Plotting program.

A Failed Calculation is a calculation which the integration program attempted to execute, but was unsuccessful. The User Interface will catch many types of illegal input at the time scenarios are defined, and will not complete execution of CALC commands if it can determine that

the scenario has undefined parameters or if the parameters given in the CALC command are illegal. However, the User Interface cannot detect all types of errors. Examples of such errors are those which might result by including inorganics or nonemitted organics in composition files used to specify compositions of ROG emissions. System errors, such as hardware faults, missing files required by the model, or lack of necessary disk space can obviously also cause failures. Failed calculations will always have integration output files (even for scenarios where "no output" is specified in the "output options"), so these files can be examined (such as by using the DISPLAY OUTP nnnn command in the MAIN mode of the User Interface, where nnnn is the calculation number) to determine the source of the error. Note that the existence of an integration program output file can be a convenient way for the user to determine whether an uncompleted calculation is still pending or whether it has failed.

10. Integration Output Files

The Integration program produces one or two main output files (in addition to updating several others), one where the calculated concentration-time data are stored in a form which can be subsequently read by the plot program (or in the PLOT mode of the User Interface), and optionally a second file which gives a printable listing of the input and the results of the calculation. Since the former file does not contain data in a form which can be printed or read directly by the user, by "integration output file" we normally refer to the printable listing file. This listing file is created by the Integration only if "integration output" is requested in the "output options" of the scenario, or if the calculation failed. Depending on the output options chosen, these files can contain the full listing of the chemical model, the concentration-time data for all species calculated, and the rates of all the reactions at all output times given. If any errors occur, these files also contain appropriate error messages. Thus, for successful calculations where the full output options are selected, these files can be quite large, and usually they are intended to be printed. Although they can be displayed on the user terminal using the DISPLAY OUTP command in the MAIN mode of the User

Interface, they are formatted to be 132 columns wide, so that the display is usually not very readable unless the user terminal has the capability to display 132 columns. (Generally, the DISP OUTP command is most useful for examining output files of failed calculations, which are generally much smaller.)

Full output should be indicated in the scenario if the user needs full documentation of the results of these calculations. However, for routine and repetitive use, such as calculations used to assess the reactivities of a large number of mixtures, this output is probably not necessary, and appropriate "plot" output can be used to document particular results of interest. The integration output files are named CALCnnnn.OUT on DEC systems and OUTnnnn on CYBER systems. Because of their size, it is important that Integration output files which have been printed or which are no longer needed be deleted, in order to save disk space.

11. Plot Output Files

Plot Output Files are the output files of the Plotting program produced when it is run as a result of the CALC command. Such files are produced when "plot output" is indicated in the "output options" of the scenario. Note that this can include tabulations of data created by the User Interface or Plotting programs, as well as page plots. These files are named CALCnnnn.PLO on DEC systems and PLOnnnn on CYBER systems, where nnnn is the calculation number for the calculation(s) created by the CALC command. These files can be displayed at the user terminal by giving the DISPLAY PLOT command in the MAIN mode of the User Interface. The user should delete Plot Output files which are no longer needed; they always can be created again by running the Plotting program, or entering the PLOT mode of the User Interface, and then using the OUT=filename option.

B. INVOKING THE PROGRAM

The User Interface and the other programs in this atmospheric photochemical reactivity modeling system are presently implemented on DEC VAX

computers running under the VMS operating system and on CDC CYBER systems running under NOS. After logging into such computers where this modeling system has been installed, and (if applicable) setting the appropriate default directory, the user can invoke the user interface program by running the command or proc file provided for this purpose. This command file will then run the User Interface program.

Upon invoking the User Interface program the number of stored scenarios, models, species names and composition files are listed, and the user is informed of the availability of the HELP feature. In addition, the User Interface will not work properly unless the keyboard terminal is set to use uppercase characters only. On most terminals there is a caps lock key that needs to be depressed (see your terminal User's Manual). The User Interface will display a reminder that you should set the caps lock on your terminal. The User Interface starts initially in the MAIN mode, and, from this mode, the user can enter the other modes by giving the appropriate commands, as documented in detail in Sections IV.C through IV.F. From any mode, the user can enter the HELP command to obtain more information on the input expected at the current mode, or, by using arguments to the HELP command, about any other mode or command in the program.

The User Interface program is terminated by giving either the EXIT or the QUIT command. If the user exits the User Interface by giving the EXIT command, the command file which was used to invoke the User Interface will then cause any calculations created by the CALC command during the session with the User Interface to be carried out. Depending on the options given to the command file, as indicated below for the two different systems, the command file will either cause the calculations to be carried out in real time, or cause them to be submitted as a batch job. This is done by executing or submitting for execution a command or proc file created by the User Interface for this purpose.

If the User Interface program is terminated using the QUIT command, then the calculations are not automatically submitted or run, nor will any command or proc file to run them be created. However, the integration input and plot input files created by the CALC command for these calculations will still exist, and the user can use them to run the Integration

program and the plot program manually by entering the appropriate system commands. This mode of operation permits the user to manually edit the Integration and Plotting program input files prior to their execution; but this is not the normal mode of operation of the User Interface, and is not documented in this manual.

The user will usually find it more convenient for the calculations to be carried out in batch mode, since running the Integration program can take significant computer time, especially if a large number of calculations have been created. The user can determine the status of batch jobs submitted by the command/proc file, or abort or cancel their execution by giving the appropriate system commands, which depend on the operating system being used. The user can also use system commands to examine any output files so produced, or can view the output files in the User Interface using the DISPLAY command.

The specific instructions for running the User Interface program as presently installed on the CYBER and the VAX computer systems are given in the following two sections.

1. CYBER Systems

The command, or proc, file used to run the User Interface on CYBER systems is called RUNSCEN. Before it can be used, it needs to be made "local." This is done by the command,

```
GET,RUNSCEN
```

which needs to be given only once in a session. Then, to invoke the User Interface, and to have any prepared model calculations automatically submitted to the system batch processor (if the EXIT command is used to terminate the program), the command to run the User Interface is:

```
RUNSCEN
```
or
```
RUNSCEN,S
```

NOTE: Once the calculations have been submitted to the batch processor, no more calculations should be prepared by the User Interface program until the batch job has begun executing.

If the user desires to have any prepared calculations to be run real time if the EXIT command is used to terminate the program, the alternative command

```
RUNSCEN,R
```

should be used to initiate the program. In either case, if the user terminates the program with the EXIT command, the command or proc file produced to run the calculations is called AFJOB.

The Integration program and Plotting program input files produced as a result of the CALC command are named INTnnnn and PLTnnnn, respectively, where nnnn is the calculation number. If the user terminated the User Interface with the QUIT command, or wants to reexecute the files after manually editing them, then he can use these files to run the Integration or the Plotting programs explicitly. To attach the Integration program to the current job (if the Integration program with this model has not already been run during the session), the user should type

```
ATTACH,Xmodelname
```

Then to run the Integration program, the user should type

```
Xmodelname
```
or
```
Xmodelname,F=nnnn
```

where "modelname" refers to the lumping model used with the scenario, which in the system presently implemented can be either M1ARM, M2ARM, LPARM, or LPBRM. The model name used is given by the MOD scenario parameter, and by the MODEL=modelname option in the integration input file. If the program is run without specifying "F=nnnn" on the command

line, the Integration program gives the prompt "NAME FOR INPUT/OUTPUT FILES =," to which the user enters "nnnn," where nnnn is the calculation number. When completed, the Integration program will give a message giving the number for the last calculation carried out, or the number of errors found, if errors occurred.

To run the Plotting program explicitly, give the commands

```
GET,XPLT
XPLT,F=nnnn
```

where again nnnn is the calculation number. (If the user wishes to run the Plotting program interactively, the F=filename option can be omitted, in which case the program operates similarly to the PLOT mode in the User Interface.) The program gives the message "PLOTS DONE" when complete.

2. VAX Systems

The command file used to run the User Interface on VAX systems is called RUNSCEN.COM. To execute this command file and cause the User Interface program to run, type

```
@RUNSCEN
```

If the user terminates the User Interface with the QUIT command, no further operations are carried out by the command file, and the control returns to the system. If the calculations are submitted to the batch processor, no more calculations should be prepared with the User Interface program until after the batch job has begun execution. Once batch processing is completed, the user should delete the AFJOB.COM file before running the User Interface program again. If the program is terminated with the EXIT command, indicating that the user wishes to have the calculations produced by the CALC command to be run or submitted, the command file will then prompt the user with:

```
S)ubmit or R)un calculations or Q)uit?:
```

The user should enter "S" to cause the calculations to be submitted to the batch processor, "R" to cause them to be run real time, or "Q" for neither. In either case, if the user terminates the program with the EXIT command, a command file to run the calculations, called AFJOB.COM is produced. (If the command file gives a prompt after the User Interface program has been terminated with the QUIT command, this means that an AFJOB.COM file from some previous session had not been deleted before running the program again. In this case, the user should respond to the prompt with "Q" to end processing.) If the user answered the above prompt with "Q," and later decides to run the calculations after all (perhaps after editing some of the input files created by the User Interface), the user can cause them to be carried out in real time by giving the command,

```
@AFJOB
```

or can cause them to be run in batch mode by giving the system command to submit the file AFJOB.COM to the appropriate batch processor. Note: The AFJOB.COM file should be used (run or submitted, and then deleted), before running the User Interface program again. If the AFJOB.COM file is not going to be used at all, it should be immediately deleted.

Note that when running the Integration or Plotting programs in real time, either by answering the RUNSCEN prompt with "R," or by giving the @AFJOB command, the programs will give prompts for file names or options. These prompts should be ignored by the user, since the information they are asking for is supplied by the command file.

The Integration program and Plotting program input files produced as a result of the CALC command are named CALCnnnn.INT and CALCnnnn.PLT, respectively, where nnnn is the calculation number. If the user terminated the User Interface with the QUIT command, or wants to re-execute the files after manually editing them, then he can use these files to run the Integration or the Plotting programs explicitly. To run the Integration program, the user should type

```
RUN INT
CALCnnnn
```

where the second command is given in response to the Integration program prompt "NAME FOR INPUT/OUTPUT FILES =." When completed, the Integration program will give a message giving the number for the last calculation carried out, or the number of errors found, if errors occurred.

To run the Plotting program explicitly, give the commands

```
RUN PLOT
OUT=CALCnnnn
IN=CALCnnnn
```

where again nnnn is the calculation number. The latter two commands are given in response to the plotting program prompt "OPTION: ." (To run the Plotting program interactively, the IN= and OUT= options can be omitted, in which case the program operates similarly to the PLOT mode in the User Interface.) The program gives the message "PLOTS DONE" when complete.

C. MAIN MODE COMMANDS AND OPERATIONS

The MAIN mode is entered when the user first invokes the program, and is the mode the user must be in to exit the program. Most commands other than those specifically involved in creating and editing scenarios or composition files can be given from this mode. These include the commands to enter the SCEN mode to create or edit scenarios, the COMP mode to create or edit composition files, or the PLOT mode to examine results of calculations. The commands which can be given in the MAIN mode are summarized in Table 5, and are documented in detail in this section.

Note that in Table 5, and in the other tables and samples of commands given in the following section, ALL CAPS refer to codes or commands which must be entered exactly as shown, brackets "[ ]" indicate optional input, and lower case indicates input which can be selected by the user. If punctuation other than brackets are shown with the command, they must be entered exactly as shown. This includes blank spaces. The symbol "<cr>"

TABLE 5. MAIN MODE COMMANDS IN THE USER INTERFACE PROGRAM.

Command	Explanation
CALC [scenario [tmc,tml]]["label"]	Causes the named scenario to be calculated. If no scenario name is given, it uses the scenario in memory. The optional "tmc" (test mixture ID composition file) and "tml" (test mixture level) given will supercede the previously defined values of TMC and TML in the scenario. The label in quotes is appended to the calculation label to aid in identifying this calculation. The scenario name, the specification of the TMC and TML, and the label are limited to a total of 48 characters.
CALCNO	Gives the calculation number for the last calculation executed (successful or otherwise) and for the last calculation created.
COMP compfilename	Enters the COMP mode to create or edit a composition file. If no composition file name is given, one is requested.
CRT	Causes program to pause during lengthy output when the screen is full (default). It also gives the user the option to abort the command causing the output, if desired.
DISPLAY OUTP calcno	Displays Integration program output on user terminal for the indicated calculation
DISPLAY PLOT calcno	Displays plots or tables for specified calculation on user terminal.
EXIT	Exits program and causes calculations created to be submitted to the system for batch processing or real-time processing. Normal termination.
EXPL	Outputs an explanatory text file.
HELP	Gives MAIN mode help display.
HELP category	Gives information about category specified.
HELP HELP	Lists all categories on which HELP is available.
INCOM	Lists the names of parameters of the scenario in memory which need to be defined before using the CALC command. (Short for "incomplete.")

TABLE 5. MAIN MODE COMMANDS IN THE USER INTERFACE PROGRAM (CONCLUDED).

Command	Explanation
LIST ALL	Lists values of all parameters for scenario in memory.
LIST parmtype	Lists values of the subset of parameters in a given parmtype for a scenario in memory. Valid parmtypes are SCEN, PHYS, CHEM, VP, NSP, OUTP, SCHEDS.
LIST ALLCOMP	Lists all available composition filenames.
LIST ALLSCE	Lists all available scenario names.
LOG [calcno]	Gives log of completed calculations. If calcno is given, lists calculations starting with the calculation specified.
PLOT	Enter the PLOT mode to plot or tabulate concentration-time profiles from integration program output.
PRINTER	Allows lengthy output to be produced without interruption. Used if the user terminal is a printer.
QUIT	Exits program without causing any calculations to be submitted or run. Integration input files are saved but the PROC file to execute calculations is deleted.
SCEN name	Enters the SCEN mode to create or edit a scenario. If no name is given, one is requested.

Brackets indicate option input.

is used in this discussion to indicate entering a carriage return only, as one might do if the default option to a prompt is chosen.

1. EXPL Command

The User Interface has two commands available to aid the user: EXPL and HELP. The first is an explanation of the User Interface program and can be viewed by entering the command of EXPL at the main prompt (i.e., MAIN: EXPL). The EXPL command will give a short overview of the program, various commands and proper syntax. The EXPL command can be used from the MAIN mode or SCEN mode in the User Interface.

2. HELP Command

The HELP command can be entered from the MAIN mode as well as any other mode of the User Interface. So when in doubt, type HELP. If no option is given in the HELP command (i.e., if "HELP" alone is entered at the terminal), then the output will be a brief list of the legal commands that can be given from the current mode. To get help in the main mode type HELP at the MAIN prompt.

The HELP command has several categories or options. The HELP HELP command will display a list of all the categories available when entered at the mode prompt. Table 6 lists all the categories on which help is available. If more information is desired for any of the these categories enter

HELP category

at the MAIN prompt, where category is one of the categories on which help is available, given in Table 6. If a category is misspelled or not available, the User Interface informs you that no help is available for this category.

The HELP command works by searching a HELP text file (named HELP.TXT on DEC systems or HELP on CYBER systems) to associate text with category names. The HELP text file is a standard ASCII file which can be

TABLE 6. LIST OF HELP CATEGORIES.

ALO	ANNOT	BMC	BML	BOC	BOL
BOS	CALC	CALCNO	CHEM	COMP	COMPFILES
CRT	DAT	DELETE	DISPLAY	DT	ENTER
ET	EXIT	EXPL	FIND	HELP	IH
IF	INCOM	LAT	LF	LIST	LOG
MOD	NEW	NORM	NOX	NO2	NSP
NXS	OUT	OUTP	PHYS	PLOT	PLU
PLT	PRINTER	PS	QUIT	RH	SA
SAVE	SCEN	SCHED	SCHEDS	SCEN PARMS	SPEC
SPO	ST	TAB	TEM	TMC	TML
TMS	TY	VAR	VP	WDY	WRA
WRM	+				

examined and, if desired, modified using the system editor. The text given at the start of the file is the text produced when HELP is given without parameters in the MAIN mode. The HELP text file in the modeling system is initially delivered with some text given to describe all of the legal User Interface commands, at a level of detail similar to that given in Table 5. However, the users can amplify or modify the contents of this file if desired in order to make it more helpful for their purposes.

3. PRINTER and CRT Commands

Upon invoking the User Interface, the default condition is that the program will pause each time the screen becomes full, at which point the user is asked if they wish to continue or quit. A carriage return will continue to display additional text and a "Q" will return the user to the current mode prompt. This is appropriate if a CRT terminal is being used. If, however, you are running the User Interface from a printer, the PRINTER command will allow continuous output of text. To return to the default terminal setting described above, enter the CRT command at the MAIN prompt. For example at the MAIN prompt you would type PRINTER to inform the User Interface you are using a printer and to return to terminal setting you would enter the CRT command which pauses when the screen is full.

4. SCEN Command

The SCEN command is used to enter the SCEN mode. The SCEN command causes a scenario to be loaded into memory and enter the SCEN mode, where scenario parameters can be modified. "Loading a scenario into memory" means that subsequent commands given in the MAIN or the SCEN mode will by default refer to that scenario. If no scenario name is given on the command line, the program will first ask if the user wants a list of existing scenarios, to which the default answer is "no." It will then prompt for the scenario name. If the scenario has not previously been defined and saved (i.e., does not exist), the program will ask for the name of a scenario to use for initializing the scenario parameters. This is helpful if the user wants to create new scenarios which are only slightly different from existing ones. A <cr> at this prompt results in most scenario parameters being undefined. The user can escape from either of these latter two prompts, and return to MAIN mode, by entering "Q" or "QUIT." The details of the operation of the SCEN mode is discussed in Section III.D.

5. LIST Command

The LIST command is actually a scenario command, and operates the same way in the MAIN mode as in the SCEN model. This command is used to list values of selected types of parameters for the scenario in memory, or to list the names of the composition files and scenarios which are available. If no scenario is in memory, this command can only be used to list names of available composition files and scenarios. The format of the command is:

LIST option

If no option is given then the User Interface will prompt for one. (The user can escape from this prompt and return to the MAIN mode by entering "Q" or "QUIT.") To display the names of all the scenarios which have been created and saved, enter

LIST ALLSCE

To display the names of the existing composition files, enter

LIST ALLCOMP

Other options for the LIST command give displays of scenario parameters, and are discussed in Section IV.D, in conjunction with the discussion of the use of this command in the SCEN mode. In this case, the parameters displayed are those for the scenario in memory (i.e., the scenario referenced last time the SCEN mode was entered.) If the SCEN mode has not been entered during this session, then there is no scenario in memory. The program will inform the user when this is the case.

6. INCOM Command

If a scenario is currently in memory, the INCOM command can be used to indicate whether any parameters of the scenario need to be defined, or whether the scenario is "complete." A "complete" scenario is one in which all parameters have been defined, and a scenario must be "complete" before a it can be used in a CALC command. This command also works in the SCEN mode.

7. COMP Command

The COMP command is used to enter the COMP mode to create or edit composition files. If the COMP command is entered without a composition file name, the User Interface will prompt the user for a file name or a question mark to list the currently available composition file names. The names of composition files can consist of 1-5 alphanumeric characters on CYBER systems, or 1-8 characters on VAX systems. If a filename is entered which does not exist, the User Interface assumes it to be a new composition file. The commands which can be given in the COMP mode are discussed in Section IV.E. Note that the operations in the COMP mode are independent of the scenario in memory.

8. CALC Command

The CALC command is used to prepare the model calculations. As discussed above, the model calculations are carried externally to the User Interface by the Integration program, based on the input files and command or proc files created by the User Interface as a result of this command. The CALC command format is:

```
CALC scenarioname tmc tml "label"
```

or

```
CALC scenarioname "label"
```

or

```
CALC "label"
```

The scenario named must be complete (i.e., have all its parameters defined -- see INCOM, above); if not, the User Interface will issue an error message and will not prepare a calculation. If the scenario name is not given, the User Interface will use the scenario currently in memory. Note that in this case, the test mixture composition file (tmc) and the test mixture level (tml) cannot be given on the command line. An error message will be given if no scenario is in memory and a scenario name is not specified

If a scenario name is given, the CALC routine uses the parameters from the disk file giving the parameters for that scenario (created as a result of the SCEN mode SAVE command), regardless of the scenario which is in memory. This is true even if the scenario named is the same as the name of the scenario in memory. Thus, if the user created a new scenario, or changed an existing one, he must SAVE it before it can be referenced by name in a calc command. Note in particular, if an existing scenario has been edited but not saved, and if the CALC command is given with that scenario named, the older version of the scenario will be used, even if the new version is the scenario in memory. On the other hand, if the CALC command is given with no scenario name, the new parameters for the new scenario will be used, since in that case it uses the scenario in memory.

If the CALC command is used with the scenario named on the command line, the user can also give a test mixture composition file name (tmc) and a test mixture level (tml) to use in the calculation. If given, these supercede the values of TMC or TML saved with the scenario. This feature is convenient for carrying out incremental reactivity calculations, where the effect of adding a given amount of test mixture or test organic is being assessed. If the TMC value is given, the TML value must also be given, or the net effect will be to have zero added test compound, which would be the same as a base-case calculation. If these values are not given, the TMC and TML values specified as part of the scenario will be used.

The user can also include a label on the CALC command line to assist in documenting this calculation. This consists of a quotation mark followed by the text to be included on the label. (A closed quotation is not required, but can be used.) This text is appended to the label of the calculation, which also includes the scenario name, and (if they were specified on the CALC command line) the TMC and TML values. The label for a calculation is included in the log file for the calculations, which the user can examine using the LOG command, and is also displayed when data from the calculation are being tabulated or plotted using the Plotting program or the PLOT mode commands. The total label, including the scenario name, the tmc and tml values (if given on the command line), and the additional label text given on the command line, should not exceed a total of 48 characters. If longer than that, the label will be truncated.

The number of calculations created by a given CALC command depends on whether there is a "varied parameter" in the scenario. If no parameter is varied, then only one calculation is created; otherwise one calculation is created for each value specified for the varied parameter. The program informs the user how many calculations are created. In addition, if the CALC command is processed successfully, the program will output the message:

***ASSIGNED CALCULATION NAME = nnnn

where nnnn is actually the calculation number for the last calculation in the set to be done. For the purpose of displaying output data, this calculation number must be known, and can be used to reference the whole set of calculations. For the purpose of plotting output data (with the PLOT command in the User Interface or running the Plotting program directly), each calculation must be referred to individually by its own calculation number. Note that when a DISPLAY OUTP or DISPLAY PLOT command is given for a series of calculations with a varied parameter, it is the number of the last calculation which must be used to reference the output for the whole series. On the other hand, when using the plotting program or the PLOT mode, each individual calculation is referenced separately by its own number.

Each time the CALC command is given, the program creates an integration program input file (INT file). In addition, if plots or tables are specified in the scenario, the program also creates a plot program input file (PLT file). When the first CALC command is issued, the program creates a command or proc file, called AFJOB.COM on the VAX, or AFJOB on the Cyber. Then, when each CALC command is given, the program writes system commands in the command/proc file for running the Integration program, running the Plotting program (if appropriate for the scenario), and routing output files to the system line printer (if appropriate for the scenario). The program also puts commands in the command/proc file to delete the INT and PLT files after they've been used, and to delete any output files that are routed to the system printer (after they've been routed, of course). This AFJOB file is executed automatically after the User Interface program is exited with an EXIT command (see Section IV.B, above).

9. CALCNO Command

The CALCNO command gives the calculation number for both the last calculation created with a CALC command, and for the last calculation successfully carried out by the Integration program. See Section IV.A.9 for a discussion of the status of calculations.

10. LOG Command

The LOG command allows the user to examine the calculation logging file created by the Integration program. For each calculation which has been successfully carried out, this file includes a record giving the calculation number, the scenario name, the calculation label, the varied parameter value (if applicable), the model name and the date the calculation was executed. The command can be given in a number of ways, as indicated below (where "n" is a calculation number).

LOG	displays entire log
LOG n	displays log from calculation number n to the end
LOG 0	displays latest calculation number
LOG -n	displays the log from the latest calculation number less n.

If the list is long and the CRT option is in effect (the default), the program will pause when the screen is full. Note also that the log file will not contain entries for calculations which are still pending, or which have failed.

11. DISPLAY Command

The DISPLAY command is used to view the integration output files or plots produced by the Plotting program at the user's terminal. This command is given as follows:

DISPLAY OUTP calcno

for integration file output or

DISPLAY PLOT calcno

for a plot file, where "calcno" is a calculation number. If the DISPLAY command is entered without any options or the calculation number, the

program will ask for them. Note that if a series of calculations were carried out in one CALC command as a result of a varied parameter, the "calcno" refers to the last calculation in the set. If a calculation number for a calculation which is not the last in the set, the program will indicate that it cannot find the requested output file. This is because the Integration and Plotting program output for all calculations created by a single CALC command are all in one file, whose name is determined by the calculation number of the last calculation of the series.

12. PLOT Command

The PLOT command is given to enter the PLOT mode, from which the user has access to many of the features of the Plotting program. This command has no arguments or options. From this mode, the user can examine plots or tabulations of results of successful calculations, given their calculation number. The commands that can be given from this mode are described in Section IV.F. The user can exit the PLOT mode by entering "Q" or "QUIT" at any of the PLOT prompts.

13. QUIT and EXIT Commands

To QUIT or EXIT commands are used to terminate the User Interface program. The differences between these commands were discussed in detail in Section IV.B. Briefly, the EXIT command is used to cause the calculations created by the CALC commands given during the session to be initiated as soon as the User Interface program is terminated, and this is the normal command used to end a successful session with the User Interface. The Quit command causes the program to terminate without causing the calculations to be carried out, and also the command or proc file used to do the calculations is not created. However, the integration input files created by the CALC command still exist. Section IV.B discusses how these calculations can be carried out manually if the program was terminated by the QUIT command. If no calculations were created, there is no practical difference between the QUIT or EXIT MAIN mode commands.

D. SCEN MODE COMMANDS AND OPERATIONS

The SCEN mode is used to edit or modify scenarios, which define all of the conditions required to do airshed model calculations other than the contents of the composition files. To enter the SCEN mode, type

SCEN scenarioname

at the MAIN prompt. All scenarios must have a name, which may consist of 1-6 alphanumeric characters (1-5 on Cyber systems). If the SCEN command is given without a scenario name, and if a scenario has been loaded into memory by a previous SCEN command, the program immediately enters the SCEN mode to edit that scenario. If the SCEN command is given with no scenario in memory then the User Interface will ask if the user wishes to list the names of the existing scenarios. If YES is entered, the list of the current scenarios will be displayed. The program then asks for the scenario name, at which point a new scenario name or an existing scenario name can be entered. If the user does not supply a scenario name, the SCEN command will be canceled. When a new scenario name is entered, a new scenario is created, and becomes the scenario in memory. However, the scenario is not saved until the SAVE command is issued, or unless the user elects to save the scenario when using the EXIT command (see below). If the user does not save the scenario before giving a SCEN or CALC command naming another scenario, or exiting the program, the scenario becomes lost.

If a new scenario is created, before any other operations can be carried out, the program will give the following prompt:

Scenario to read (? to see old filenames, <cr> if none):

The purpose of this prompt is to let the user initialize all the scenario parameter values with those from an existing scenario, which is useful if the new scenario is to differ in only a few respects from an existing one. A list of existing scenarios can be obtained by entering "?." If the user names a nonexistent scenario, the program will list all the

existing scenarios, and repeat the above prompt. (A "Q" or "QUIT" will cancel the SCEN command, and return to the MAIN mode.) If a <cr> is entered, the new scenario has all its parameters undefined. (In this case, the user might find the ENTER command, which prompts for values for all scenario parameter, convenient. However, he can also define all scenario values using commands for each parameter.) All parameters must be defined (i.e., the scenario must be "complete") before the scenario can be used in a CALC command, though a scenario does not have to be complete to be saved.

A list and description of the all the scenario parameters used in this model are given in Table 7. These parameters can have various types of values, either a number, a code, a composition file name, etc., and the table indicates the types of values required for each parameter. These parameters are defined or modified either through the ENTER command, or by giving the parameter name and the new value, as discussed below. A summary of the SCEN mode commands is given in Table 8. The SCEN mode commands, and the scenario parameters and options which need explanation beyond that given in Table 7, are discussed in the remainder of this section.

1. LIST Command

The LIST command in the SCEN mode operates the same way as it does in the MAIN mode. This command is used to list the current values of selected groups, or all, of the scenario parameters. For example, the ALL option of the LIST command gives the values of all the parameters for the scenario in memory, while, for example, LIST CHEM will only give chemical parameters values. In addition, as indicated above in the discussion of the LIST command in the MAIN mode, this command can also be used to list the names of all of the scenarios or all of the composition files which have been created and saved. The format of this command is

LIST option

TABLE 7. LIST AND DESCRIPTIONS OF ALL SCENARIO PARAMETERS.

Code	Value type[a]	Description
		General Scenario Parameters (SCEN)[b]
MOD	model	Chemical model name. The chemical model determines how the reactions of the chemical species are represented. As initially implemented, legal model names are LPARM, LM1ARM, LM2ARM (M1ARM and M2ARM on Cyber systems), and LPBRM. These are discussed in Section III.
VAR	parm	Name of the varied parameter, if any. Parameters which can be varied are LF, IH, RH, ALO, or NOX. See Section IV.A.5 and IV.D.6.
PLU	code	Pollutant Level Input Units (MOLAR or PPM). This determines units for the NO_x and ROG emissions levels, i.e., NOX, BOL, and TML. If PLU=MOLAR, then these are given in millimoles m^{-1}, and thus,refer to total integrated emissions fluxes per unit area. If PLU=PPM, then NOX, BOL, and TML are given in units of ppm at the initial inversion height. See Section IV.D.5.
		Physical Parameters (PHYS)[b]
DT	y/n	Daylight Time used? If "yes," then one hour is subtracted from the clock time to determine the solar time for the purpose of calculating the photolysis rates.
LAT	number	Latitude in degrees. The latitude affects the calculated solar light intensities used in calculating the photolysis rates.
DAT	number	Date, given as "month.day," e.g., 6.07 = June 7. This determines the time of year, which also affects the light intensities and thus the calculated photolysis rates.
ST	clock	Start Time for the simulation. The simulation is usually started early in the day, or in the early morning before the emissions start.
ET	clock	End Time for simulation. Must be later then ST. If a multiday simulation is to be carried out, also give the day number. For example, to end the simulation at 9 AM the following day, enter ET=10900.

TABLE 7. LIST AND DESCRIPTIONS OF ALL SCENARIO PARAMETERS (CONTINUED).

Code	Value type[a]	Description
SA	number	Time Interval to save data (minutes). This determines how often the integration program saves or prints the calculated concentrations.
LF	nbr/sch	Light Factor. This factor is multiplied by the theoretically calculated photolysis rates to derive the values to use in the calculation. Use LF=1.0 for full sunlight, LF<1 for overcast conditions. LF can be scheduled to represent the effects of passing clouds. (Default = 1).
IH	nbr/sch	Inversion Height in meters. This important parameter determines the total mixing volume into which the pollutants are emitted. Normally it is scheduled to represent the effect of the mixing height increasing throughout the day.
TEM	nbr/sch	Temperature, in degrees K. This is used to determine the rates of the temperature-dependent chemical reactions. (Default = 300 K).
RH	number	Relative Humidity in percent (100.0 = saturated). This is used to determine rates of chemical reactions where water is a reactant. (Default = 50 percent).
		Chemical Species Parameters (CHEM)[b]
BMC	p.cmp	Background Mix Composition Filename. This file contains the chemical species present initially in the simulation, besides those which are emitted. The concentrations of the species in the composition file must be in ppm.
BML	number	Background Mix Level factor (Default = 1.0). All the concentrations given in the BMC file is multiplied by this factor to give the background levels actually used in the simulation. To use the values in the file as they are, use the default.
ALO	p.cmp	Aloft Mix Composition Filename. This file contains the chemical species which are represented as being in the aloft air mass, which are entrained into the air mass being simulated if the inversion height raises. If the inversion height is constant (i.e., not scheduled) or only decreases with time, the ALO parameter has no effect

TABLE 7. LIST AND DESCRIPTIONS OF ALL SCENARIO PARAMETERS (CONTINUED).

Code	Value type[a]	Description
		on the simulation. All concentrations given in the ALO file must be in ppm.
NOX	number	Total amount of NO_x emitted, in ppm or millimoles m^{-2}, depending on PLU (see Section IV.D.4). Note that this does not count NO_x which might be in the background (BMC) or aloft (ALO) mixture.
NXS	code	NO_x Schedule. Enter SCHED if NO_x is emitted (even at a constant rate), or NONE if all NO_x is present initially. See Section IV.D.4 and IV.D.5
NO2	number	NO_2/NO_x ratio. "NO_x" consists of both NO and NO_2, and this parameter determines the relative amounts of each. An error message is given if a NO_2 value out of the range 0.0 - 1.0 is given
BOC	e.cmp	Base Organics Emissions Composition Filename. This file must contain only legal ROG species names, and it should be normalized to 1 ppmC total. If the file contains inorganic or other non-emitted ROG species, the User Interface will not detect this, but the calculation will fail at integration time. If the concentrations in the file are not normalized, then the specified BOL value will not reflect the correct amount of ROG carbon emitted. See Section IV.D.5.
BOL	number	Total amount of Base Organics emitted, in ppmC or millimoles C m^{-2}, depending on PLU (See Section IV.D.5. This does not count any organics which might be present among the background (BMC) species.
BOS	code	Base Organics Schedule. Enter SCHED if the base organics are emitted, even if at a constant rate, or NONE if all are present initially. See Section IV.D.4 and IV.D.5.
TMC	sp/ecmp	Test Mixture Composition. This can be either a composition filename, or the name of an individual ROG species. If a filename, it must contain only legal emitted ROG species names and it should be normalized, as with BOC files. If an ROG species name (i.e., a name listed in Table 2), the effect is the same as referencing a composition file with the named species present at a level of 1.0. (Note that in this case, TML will refer to moles or ppm emitted, rather than moles C or ppmC, as would be the case if TMC

TABLE 7. LIST AND DESCRIPTIONS OF ALL SCENARIO PARAMETERS (CONTINUED).

Code	Value type[a]	Description
		referred to a normalized composition file.) This parameter can be superseded by the CALC command, as discussed in Section IV.C.8.
TML	number	Total amount of Test Mixture emitted, in ppmC or millimoles millimoles C m^{-2}, depending on PLU (See Section IV.D.4.) Note that the total amount of emitted organics in the scenario is BOL+TML. This parameter can be superseded by the CALC command, as discussed in Section IV.C.8. See also Section IV.D.5.
TMS	code	Test Mixture Schedule. Enter SCHED if the base organics are emitted, even if at a constant rate, or NONE if all are present initially. See Section IV.D.4 and IV.D.5.
		Varied Parameter Values (VP)[b]. See Section IV.D.6.
VP	value	Add new value for the varied parameter. The value must be consistent with the type of parameter which is varied, though the User Interface does not check for this. If an inappropriate type of value is entered, the calculation will fail at integration time. Note that the varied parameter values can be entered in any order, and that the program does not check for duplicate values. However, it is recommended that the values be ordered, since the ordering of the VP values determines the order that the calculations are carried out.
VPn	value	Replace the nth varied parameter value with the new value. E.g., VP2=value replaces the second value on the list of varied parameter values. The LIST VP will show the codes to use for each value.
VPn,X		Delete the nth varied parameter value. All subsequent values are moved down by one; i.e., the n+1th parameter becomes then nth, etc. The LIST VP will show the codes to use for each value.
		Nonstandard parameters (NSP)[b]. (See Section IV.D.7)

TABLE 7. LIST AND DESCRIPTIONS OF ALL SCENARIO PARAMETERS (CONTINUED).

Code	Value type[a]	Description
NSP	parm,[number]	Add a new nonstandard parameter named "parm." If it is a parameter which requires a numerical value, the user should enter this also. This can be used, for example, to change a rate constant or the value of a mechanistic parameter in the chemical model. No checking is done by the User Interface to determine if the parameter name is one that will be recognized by the Integration program, whether a numerical value is required, and whether its magnitude is appropriate. Invalid nonstandard parameters will cause the simulation to fail at integration time. A knowledge of the details of the model and the features of the Integration program is required to use this option. Note that the program does not check for duplicate nonstandard parameter names.
NSPn	parm,[number]	Change nth nonstandard parameter and value. This works analogously to VPn and PSn. If no numerical value is given and the previous nth parameter had a value, the previous value is deleted. Note that to change the value, it is also necessary to reenter the parameter name. LIST NSP gives the codes associated with each nonstandard parameter.
NSPn,X		Delete the nth nonstandard parameter and value. All subsequent parameter and value entries are moved up by one. LIST NSP gives the codes associated with each nonstandard parameter.
		Output Options (OUTP)[b]. See Section IV.D.8.
OUT	y/n	Produce integration output listing file? If "no" no Integration program listing file is produced unless an error occurred at integration time, and the WDY, WRA, and WRM options are ignored. Default = "Yes." See Section IV.A.11.
PLT	y/n	Produce plots of calculated results? If "yes," the CALC command will also produce input and command files to cause the Plot program to produce a plot output file giving plots of the model species specified by the PS command. (Default = "No").

TABLE 7. LIST AND DESCRIPTIONS OF ALL SCENARIO PARAMETERS (CONTINUED).

Code	Value type[a]	Description
TAB	y/n	Produce tabulations of calculated results? If "yes," the CALC command will also produce input and command files to cause the Plot program to produce a plot output file giving tabulations of the model species specified by the PS command. If both PLT and TAB are selected, then the plot output file will contain both plots and tabulations of the PS species. (Default = "No").
WDY	y/n	Summarize derivatives of concentrations on output listing? If "yes" the rates of change of the calculated concentrations of the non-steady-state model species are included in the Integration program output listing file for all the times for which the concentrations are printed and saved. Used only if OUT=Y. (Default = "No").
WRA	y/n	Summarize rates of reactions on output listing? If "yes," the rates of all the reactions in the mechanism are included in the integration output listing file for all of the times for which concentrations are printed and saved. Used only if OUT=Y. (Default = "No").
WRM	y/n	Produce full model listing in integration output file? If "yes", a listing of the chemical mechanism in the model used, including lists of the reactions, model species, and variable coefficient values, is included at the beginning of Integration program output file. Used only of OUT=Y. (Default = "No").
PS	mod.spc	Add the model species named to the list of species to be plotted or tabulated. Used only if PLT=Y. Note that for emitted ROG species, the species named here must refer to the model species used to represent them in the model, which in general are different from those used to specify ROG species in composition files. A list of legal model species names, which can be referenced as PS values, is given in Table 3. No effect if both PLT and TAB are "no."
PSn	mod.spc	Modify the nth compound on the list of species to be plotted or tabulated with the one indicated. Note that if a plotted species value is deleted by the PSn,X command, all the subsequent values are moved down by one. The LIST OUTP command will show the codes to use for each value.

TABLE 7. LIST AND DESCRIPTIONS OF ALL SCENARIO PARAMETERS (CONCLUDED).

Code	Value type[a]	Description
PSn,X		Delete the nth compound from the lsit of species to be plotted or tabulated. All the subsequent species on the list are moved up by one. The LIST OUTP command will show the codes to use for each value.

[a]Types of legal value for these parameters are as indicated by the following codes:

code = One of several codes appropriate for this parameter. Legal codes indicated.

clock = Clock time, in format HHMM; i.e., 1330 for 1:30 PM. For times on subsequent days, format is DHHMM, where D=1 means next day, D=2 means day after that, etc.

nbr/sch = Either a number (units indicated) or the code "SCHED" to indicate that this parameter has a time varying schedule.

mod.spc = Model species name.

number = A single real number. If the number has required units, these are indicated.

model = Model file name.

parm = Scenario parameter name.

p.cmp = Pollutant composition file. Concentrations of of constituents in file must be in ppm.

e.cmp = ROG emissions composition file name. (Should contain only ROG species and be normalized).

sp/ecmp = ROG species name or emissions composition file name.

value = Parameter value. Depends on type of varied parameter.

y/n = Yes or No.

[b]Parameter group code as used by the LIST command.

TABLE 8. SUMMARY OF SCEN COMMANDS IN THE USER INTERFACE PROGRAM.

Command	Explanation
parameter[=value]	Changes the parameter to the new value. If no value is given, the program gives the current value and prompts for a new one. A <cr> at that prompt leaves the current value unchanged. See Table 7 for the list of parameters.
DELETE	Deletes saved scenario. Program asks for scenario name to verify delete.
ENTER	Enters mode where program prompts for values of all scenario parameters.
EXPL	Outputs the explanatory text file. Same as EXPL in MAIN.
EXIT	Returns to MAIN after asking if the user wants to save scenario.
HELP	Gives SCEN mode help display.
HELP category	Gives information about category specified.
HELP HELP	Lists all categories on which HELP is available.
INCOM	Lists the names of parameters of scenario which which need to be defined before using the CALC command.
LIST ALL	Lists values of all parameters for scenario in memory.
LIST parmtype	Lists values of the subset of parameters in a given type. Parameter types are SCEN, PHYS, CHEM, VF, OUTP, and NSP. Parameters associated with each type are shown in Table 7.
LIST SCHEDS	Display all schedules which have been defined.
LIST SCHED parm	Display the schedule for the specified scheduled parameter. Parameters which could be scheduled are IH, LF, TEM, ALO, NXS, BOS, or TMS.
LIST ALLSCEN	List names of all available stored scenarios. Also produces a LIST SCEN output.

TABLE 8. SUMMARY OF SCEN COMMANDS IN THE USER INTERFACE PROGRAM (CONCLUDED).

Command	Explanation
LIST ALLCOMP	List names of all available stored composition files.
NEW	Deletes current scenario in memory. First asks is the old one to be saved. Then allows the user to create or edit another scenario.
QUIT	Exits SCEN mode and returns user to the MAIN mode. Scenario remains in memory, but is not saved on disk.
SAVE	Saves the newly edited scenario on disk, and returns to MAIN mode. Scenario remains in memory.
SCHED parm	Enter mode to edit or define a time schedule for a parameter. Parameters which can be referenced are LF, IH, TEM, NXS, BOS, or TMS.

Brackets indicate optional input.

If an option is not given, the program will prompt for one. (Entering a "Q" or "QUIT" will allow the user to escape from this prompt.) The ALL option gives a listing of all of the current scenario values. The SCEN, PHYS, CHEM, VP, OUTP, and NSP options give listings of a subset of the parameters, as indicated in Table 7. The SCHED option gives listings for the current values of the schedules for the time-varying parameters.

In addition, the LIST ALLSCE gives a listing of all the scenarios which have been saved, and the list ALLCOMP gives a listing of all the available composition file.

2. ENTER Command

The ENTER command is used to have the program request values for all the scenario parameters, one after the other. This is convenient if all or most of the parameters need to be changed, and may be more

convenient for new users, since it does not require memorizing all the parameter codes (other than those for entering schedules). After this command is entered. the program goes through each parameter (in the order as given in Table 7), giving the current value, and prompting for a new one. At each prompt the user can either:(1) hit <cr> to leave the value unchanged, unless the value is undefined, (2) enter a new value, (3) remove a previously defined value by entering "X," (4) ask for help by entering "HELP," or (5) quit the ENTER mode by entering "Q" or "QUIT." Note that if the user QUIT's the ENTER mode, the changes that were made to the previous parameters when in this mode are kept.

To enter a new value with the ENTER command, just type in the desired value at the prompt. If a value currently exists the new value will replace the old one. If an illegal value is entered, or the parameter is undefined, the User Interface will not allow you to continue until a correct response is entered. (If the parameter is undefined and the user wishes to keep it that way for now, he must enter "X.") The Aloft Mix ID, Background Mix ID, Base Organics Comp ID and Test Mixture ID require filenames; if the user is not familiar with these file names a ? should be entered, which produces a list of all the legal file names.

To remove the values for most parameters, enter an X at the prompt. This will either reset the value to U or UNDEF for undefined, or to no value, depending on the parameter. The X command will not work on the latitude (LAT) and date (DAT) parameters. Therefore, some value must be entered.

The program may also enter the SCHED submode for the scheduled parameters. However, it does not give prompts to lead the user through entering the schedule data required. The SCHED submode commands are discussed in Section IV.D.5.

3. Editing Parameters

In order to define or change the value for an individual scenario parameter, one needs to reference it by its two- or three-letter code. These code names, are given along with the parameter descriptions in in Table 7. Any of these parameters can be changed using the following

general command:

parm=value

or parm,value

or parm value

or parm

where "parm" refers to one of the parameter codes in Table 7. To set a currently defined parameter as undefined, enter

parm,X

However, the DAT and LAT parameters cannot be deleted.

If the parameter code is entered with no value, the program will show the current value and then prompt the user for a value. In this case, the user can either:(1) give a new value, (2) enter <cr> to leave the parameter unchanged and return to SCEN mode (provided that the parameter has a value, and is not undefined), (3) enter "X" to delete the value and make the parameter undefined, or (4) type "HELP" to get the HELP display for this parameter. The user can get the same HELP display for the parameter by typing "HELP parm," where "parm" is the parameter code.

4. Schedules

Schedules are used to specify how certain types of physical parameters vary with time, and are also required for specification of emissions. The parameters which can vary with time, and thus be "scheduled" are the inversion height (IH), the light intensity factor (LF), the temperature (TEM), and the NO_x, (NXS), base ROG (BOS), and test ROG (TMS) emissions. The ways in which the schedules are entered are similar regardless of for which parameter the schedule is being defined.

although there are special considerations in the case of emissions schedules. In this section, we discuss the commands and considerations that are common to all types of schedules. There are several special considerations in emissions schedules, and these are discussed in the following section.

Before a schedule can be entered for a given type of scenario parameter, it is first necessary to declare it as scheduled. In the case of the physical parameters (IH, LF, or TEM), this is done by giving the parameter the value "SCHED." For example, if in a scenario one wishes to have the inversion height constant at 300 meters, one would enter

IH 300. (or IH,300 or IH=300)

while if it is to vary with time, one would enter instead

IH SCHED (or IH,SCHED or IH=SCHED)

and then use separate SCHED submode commands, discussed below, to define how it varies. In the case of emissions, there are separate parameters which are used to indicate whether schedules are used: NXS for NO_x emissions, BOS for background ROG emissions, and TMS for base case emissions. These parameters can take on the values "SCHED" if the associated emissions are scheduled, or "NONE" if not. Thus, for example, if NO_x emissions are to be scheduled, one would enter

NXS SCHED (or NSX,SCHED or NXS=SCHED)

The parameters NOX, BOL, and TML cannot be given the value SCHED in a manner analogous to the usage with the scheduleable physical parameters because, as discussed in the following section, their values are required even when they are scheduled.

Once a parameter has been declared as scheduled, its schedule can be defined by entering the SCHED submode for that parameter. This is done by the following SCEN mode command:

SCHED parm

where parm can be IH, LF, TEM, NXS, BOS, or TMS. If the parameter named has been declared as scheduled, it responds with the prompt "SCHED parm:," to tell the user that he can now edit or define the schedule for the parameter named. (If the parameter has not been scheduled, of if a non-scheduleable parameter is named, the program will give an error message.) The commands which the user can enter in the SCHED submode are summarized in Table 9.

TABLE 9. SUMMARY OF SCHEDULE COMMANDS.

Command	Explanation
LIST	Lists the schedule and all schedule parameters.
TY=code	Specifies whether this is a linear or a stepped schedule. Legal codes are "L" for linear or "S" for stepped. This must be defined for each schedule.
IF=code	(For emissions schedules only.) Gives the fraction of the pollutants present initially, as opposed to the being emitted. Must be between 0 and 1.
time=number	Gives the value for the parameter at the clock time indicated. The time is given as clock time, i.e., DHHMM, where HH = hours (24 hour time), MM=minutes, and D = day number following the first day, if a multi-day simulation. These can be entered in any order; the program automatically sorts them in order of ascending time. If there is already an entry in the schedule for this time, this command results in a new value being given for that time.
time=X	Deletes the schedule entry for the time indicated. Must be a time already defined. Again, time is in clock time.
NEW	Deletes all time points in the schedule. However, the values of the schedule type (TY) and (for emissions schedules) the initial fraction (IF) are unchanged by this command.
<cr>, Q, or QUIT	Returns to SCEN mode.

Note that the schedule type (TY) parameter must be defined for all schedules used, or the scenario will be incomplete. This parameter determines how values of the parameter are calculated for times intermediate to those explicitly entered in the schedule. If the type is "linear" (TY=L), then the values for the intermediate times are calculated by linear interpolation between the times specified. If the type is "stepped" (TY=S), then the values are constant at the value given at the previous time, and change in a stepwise manner at the times where new values are given. For example, if the temperature schedule contains the entries

0900 290.
1500 310.

then the temperature at noon would be 300 K if the type is linear, but would be 290 K if it is stepped. In the latter case, the temperature would change abruptly from 290 K to 310 K at 3 PM.

All schedules must also have at least two entries where a value is given for a specified time. These are entered or edited as shown in Table 8. For schedules for physical parameters (IH, LF or TEM), the values given must be in the appropriate units for that parameter, i.e., meters for IH, a unitless factor for LF, or degrees K for TEM. For emissions schedules, the values entered refer to relative emissions fluxes, as discussed below. The times and values can be entered by the user in any order; the program will automatically sort the entries in order of ascending time.

For schedules of time varying physical parameters, it is not necessary that the first and last time for which a value in the schedule correspond to the starting and ending time of the simulation. (As indicated in Table 7, the starting and ending times of the simulation are determined by the ST and the ET scenario parameters, respectively.) If the first time in the schedule is after the start time of the calculation, then the value of the parameter for the times up to the first time given on the schedule is constant at the first value given. If the last time on the schedule is before the end of the calculation, then the value after

the final time in the schedule is constant at the last value specified. If the schedule for the parameter is given for times outside the time range of the simulation, the entries which do not affect the time period of the simulation are ignored, with the initial and/or the final values of the parameter in ths simulation being determined by linear extrapolation or by the value defined for a previous time, depending on whether the schedule is linear or stepped.

On the other hand, for proper calculation of emissions rates, the initial and final entries in schedules for emissions normally should correspond exactly to the initial and final times of the simulations. The only exceptions to this are (a) if the initial entry is for a time after the start of the simulation, and the relative emission rate given for the initial entry is zero, or (b) if the final entry is for a is for a time before the end, and it is also zero. This is because, as discussed in the following section, the entries in emissions schedules refer to relative emission fluxes, and are normalized according to the area underneath the emissions vs. time curve defined by the emissions schedule. In calculating the area under this curve, the program starts at the first time in the schedule, and ends at the last time of the schedule, regardless of the initial and the final time of the calculation. Thus, if the emissions schedule indicates nonzero emissions for times before or after the emissions, the area under the total emissions vs. time curve defined by the schedule would include a contribution for a time period not being simulated, and thus the total integrated pollutant input during the simulation would be less than specified by the corresponding pollutant level (NOX, BOL, or TML) parameter. Note that the program does not check for this, and, in some applications, violating this rule might be appropriate. This is discussed further in Section IV.D.5.c.

Emissions schedules have an additional parameter, IF, which determines the fraction of the pollutants which are initially present. If this value is 1.0, then all pollutants are present initially, and there are no emissions. In this case, the time and value emissions entries are not used, but at least one entry must still be given. If the initial fraction, IF, value is 0, then no pollutants of this type are present initially (other than those specified in the background mixture, if any),

and they are all emitted. Note that the value of the IF schedule parameter does not affect the total pollutant input (this being determined by the NOX, BOL, or the TML parameters), but just how much is present initially as opposed to being emitted. This and other special factors involved with emissions schedules are discussed in the following section.

As indicated in Table 9, the user enters a <cr>, "Q," or "QUIT" at the SCHED prompt to return to the SCEN mode. There is no way to exit the SCHED submode and abort the changes made to the schedule since entering it, since all editing operations are made directly on the values stored in memory.

Note that if the user removes a previously defined schedule from a scenario by setting a numerical value to IH, LF, or TEM, or by setting NXS, BOS, or TMS to "NONE," the values given for the previous parameter are not deleted, but they become invisible to the user. This is true even if the scenario is saved and then later reread. If the parameter is "re-scheduled," the previously defined values reappear. The only ways the entries for a schedule can be completely deleted is by giving the "NEW" command at the SCHED prompt for the schedule whose values are being deleted.

5. Specifications of Emissions

The specifications of emissions involve specifying three types of scenario parameters, those defining the composition of the species emitted, those defining the total amounts emitted, and those defining the emissions schedule. The types of emissions which are represented separately in this model are: (1) NO_x emissions, whose composition is determined by the NO2 parameter and whose amounts are determined by the NOX parameter, and whose schedule is determined by the NXS parameter; (2) Base case ROG emissions, whose composition, amounts, and schedule are determined by the BOC, BOL and BOS parameters, respectively; and (3) test ROG emissions, whose composition, amounts and schedules are determined respectively by the TMC, TML and TMS parameters. The meaning of these parameters are summarized in Table 7, and the means of entering and modifying them were discussed in the previous sections. Special factors

which need to be taken into account in the specifications of emissions, particularly with regard to emissions levels and schedules, are discussed in this section.

a. Emissions Compositions

The NO2, BOC and TMC parameters are all similar in that they are used to specify compositions, rather than amounts, of the various types of emissions. In the case of NO_x emissions, in this model, only two species, namely NO and NO_2 are included in this group when the model is run under the control of the User Interface. Therefore only one numerical parameter, NO2, which gives the ratio of NO_2 to NO_x emissions, is required to totally specify the composition of this class of emissions. In the case of the base case and the test ROG emissions, a number of species can be emitted, and thus, composition files are employed to specify the relative amounts of each. In the case of test mixture compositions, the name of a ROG species or class, such as (for the current version of the model) one listed in Table 2 can also be used to indicate that the test ROG emissions are a pure compound (or all in one class of compounds, such as, for example, monoalkylbenzenes) rather than a mixture.

When composition files are used for specification of emissions compositions, it is important that they be "normalized" so that they reflect only relative compositions. A "normalized" composition file is one such that when the amounts given for all the species are multiplied by the number of carbons and summed up, the total comes to 1.0. (Another way of describing it is that if the composition files are thought of as giving amounts of each ROG species in units of ppm, ppm, then the total "concentration" in of all the ROG species in the file would sum up to 1 ppmC.) For example, if a test mixture contains equal molar amounts of n-hexane (6 carbons), propene (3 carbons), and m-xylene (8 carbons), then in order to be normalized, the composition file must be as follows,

```
N-C6      0.0625
PROPENE   0.0625
TOLUENE   0.0625
```

in order to sum up to one carbon. Note that the program does not automatically check for proper normalization of composition files, nor will the calculation fail if the file is not properly normalized. (The consequence of using nonnormalized files is that the BOL or the TML parameter would not accurately indicate the total amounts of base or test ROG emitted.) Assuring normalization of composition files for ROG emissions is most conveniently done by using the NORM command in the COMP mode, as discussed in Section IV.E.3, below.

b. Emissions Levels

The total amounts of each of the three types of emissions are indicated by the NOX, BOL, and the TML parameters. In each case, they are used to indicate the total amount of each which are initially present or emitted throughout the simulation. The units in which these emissions are expressed depend on two factors: (1) whether the emissions of this group of pollutants are "scheduled" (see following section), and (2) if they are "scheduled," on the value of the PLU parameter. Note that the specifications of NO_x, base case ROG, or test ROG emissions amounts and schedules are independent of each other, and thus, it is possible, for example, for one group to be scheduled and one not (though it is recommended in most cases that a consistent method to specify emissions amounts and units be employed). The factors involved for all three types are essentially the same, and are discussed below.

If a class of emitted pollutants is to be represented as being emitted over time, rather than all being present initially, it is necessary that its emissions be "scheduled." This is the case regardless of whether the emissions rates are assumed to vary with time, or to be constant throughout the simulation. Note that emissions schedules allow for the possibility for a fraction (or even all) of the pollutants to be present initially, with the remainder being emitted, so use of schedules allow for maximum flexibility. Whether the NO_x, base ROG, or test ROG emissions are scheduled or not are determined by the values of the NXS, BOS, or TMS parameters, respectively. Each of these parameters can have a value of "SCHED," indicating that emissions are scheduled or "NONE," indicating that they are not.

Regardless of whether the emissions are scheduled, the units used to define the quantities emitted are determined by the PLU parameter, which can have a value of either "M," for "Molar," or "P," for "Ppm." If PLU=M(olar), then the the emissions amounts are expressed in terms of integrated emissions fluxes, i.e., the total amount emitted per unit during the simulation. The flux units required by this model are always given in terms of millimole m^{-2} or millimole C m^{-2}. Note that in this case, the total amount of moles (or moles Carbon) introduced in the air mass in a simulation is independent of the volume of the air mass (as determined by the inversion height, or IH, parameter). On the other hand, the concentrations resulting from the emissions would be affected by the inversion height, with lower inversion heights (i.e., lower volumes of the air mass) resulting in higher concentrations. Since the output of emissions sources are generally given in terms of emissions fluxes, and are obviously not directly affected by the inversion height, this is usually the most convenient way in which to express emissions in most airshed simulations.

If PLU=M(olar) and the emissions schedule is such that a certain fraction, or all, of the emissions are present initially, then the initial amounts of the pollutants present are given in terms of a certain amount of millimoles per square meter. This is converted to ppm concentration units based on the inversion height and the temperature, using the perfect gas law. For example, if the integrated NO_x fluxes (i.e., the NOX parameter) = 2 millimoles m^{-2}, and the initial fraction (IF) in the NOX schedule (see below) is 0.25, then 0.5 millimoles of NO_x will be present in the initial air mass over each square meter of area. If the initial inversion height is 100 meters, and the temperature is 300 K, then the initial NO_x concentration is given by

$$\begin{aligned}\text{Initial } NO_x &= (10^6 \text{ ppm atm}^{-1}) \times (0.5 \times 10^{-3} \text{ moles } NO_x \text{ m}^{-2}) \\ &\quad \times (8.2056 \times 10^{-5} \text{ m}^3 \text{ atm deg}^{-1} \text{ mole}^{-1}) \\ &\quad \times (300 \text{ deg}) / (100 \text{ m}) \\ &= 0.123 \text{ ppm.}\end{aligned}$$

where 8.2056×10^{-5} factor is the gas constant for the units indicated.

The NO_x emission rates for the above example would be calculated based on the NXS emissions schedule such that the total integrated amount of NO_x emitted during the simulation would be the remaining 2.5 millimoles m^{-2}. Thus, if in this example the NO_x emissions are assumed to be constant from the start of the simulation and then end abruptly 5 hours (300 minutes) later, then the emission flux of NO_x would be,

$$NO_x \text{ flux} = (2.5 \times 10^{-3} \text{ mole m-2}) / (300 \text{ min})$$

$$= 8.33 \times 10^{-6} \text{ moles } m^{-2} \text{ min}^{-1}$$

and the rate of change of NO_x due to emissions at time t would be

$$d[NO_x]/dt \text{ (ppm min}^{-1}) = (8.33 \times 10^{-6} \text{ moles } m^{-2} \text{ min}^{-1}) \times (8.2056 \times 10^{-5} \text{ m}^3 \text{ atm deg}^{-1} \text{ mole}^{-1}) \times (10^6 \text{ ppm atm}^{-1}) \times T(t) / H(t)$$

where T(t) and H(t) are the temperature (in deg K) and the inversion height (in meters), respectively, at time t. Note that since both the temperature and the inversion height could vary with time, the rate of change of NO_x due to emissions could also vary, even if the NO_x emission flux is constant. The emissions schedules always refer to emissions fluxes, regardless of which pollutant level units (PLU) are used.

If PLU=P(pm), then the amounts emitted are expressed in terms of the resulting concentration (in ppm or ppmC) of the pollutants which would occur if they did not undergo reaction. If this option for emissions units is chosen, then emissions fluxes will depend on the inversion height. In order to avoid ambiguity if the inversion height varies with time, if PLU=P, if the inversion heignt varies with time (i.e., if IH is scheduled), emissions fluxes are always calculated based on the <u>initial</u> inversion height, regardless of how it varies subsequently. Thus, for example, if PLU=P, and if the emissions schedule for NO_x specifies that 0.25 of of the NO_x is present initially with the remaining being emitted at a constant flux for the first 300 minutes of the

simulation and none thereafter, and the NO_x emissions level parameter, NOX, = 0.3, then the initial NO_x concentration is 0.075 ppm, and the rate of change in NO_x at time t due to emissions is given by

$$d[NO_x]/dt \text{ (ppm min-1)} = [\ (0.0225 \text{ ppm}) / (300 \text{ min.})\ ] \times [\ H(0)/H(t)\ ]$$

where H(0) is the initial inversion height, and H(t) is the inversion height at time t.

Note that if the emissions for a certain group of pollutants are not scheduled, then all the pollutants in the group are assumed to be present initially. Thus, the effect of having the emissions not being scheduled is the same as the emissions being scheduled, but with the initial fraction (IF) set to 1.

The treatment of emission levels of base or test ROG species is totally analogous to that for NO_x, except that in this case, emissions are generally expressed in terms of mole carbon or ppmC, rather than simple molar or ppm units as is the case for NO_x. This is, of course, provided that the composition files used for base and test ROG emissions are properly normalized. The one exception to this is the case when the name of an individual ROG species or class name is given to specify the test mixture composition (TMC) rather than the name of a composition file. In this case, the test mixture level (TML) parameter refers to the number of moles (or ppm) of the pollutant specified. If the user desires to have the TML units for single-component test mixtures be in mole carbon (or ppmC) units for consistency with the usage for mixtures, he must create a normalized composition file containing the single species. Note that the file must be given a different name than the species name; the COMP command will not allow composition file names to be the same as a legal ROG species name.

c. Emissions Schedules

Except for the addition of the "initial fraction," IF, parameter, emissions schedules are entered and edited in exactly the same

way as discussed in Section IV.D.5 for the schedules for the physical parameters. However, unlike the entries in the schedules for the physical parameters, the entries in the emissions schedules refer to relative emissions fluxes, and indicate only how emission fluxes vary with time, and not the amounts emitted. This permits the NOX, BOL, or TML parameters to refer to the total amount of the various types of pollutants introduced into the air mass, regardless of the details of its schedule. The specific emissions fluxes are calculated such that the total integrated emissions into the air mass, plus the pollutants present initially (if the initial fraction, IF, is nonzero) is equal to the value given by emissions level parameter, as discussed above in the previous section.

An example can be used to illustrate specifically how absolute emissions rates are calculated based on the relative entries in an emissions schedule. If PLU=MOLAR, NOX=3, and the entries in SCHED NXS are

```
TY=L
IF=0.25
0800 0.0
0900 1.0
1500 1.0
1600 0.0
```

then the a total of 3 millimoles of NO_x is introduced into the air mass for each square meter of surface area, of which 0.75 millimoles are present initially, and 2.25 millimoles are emitted between 0800 and 1600 at rates proportional to the relative fluxes indicated above. Since the area under the relative emission flux vs. time curve given above is 420 min, then in order to yield the an integrated emissions flux of 2.25 millimoles m^{-2}, the above relative values are multiplied by a factor of (2.25 millimoles m^{-2})/(420 min) to obtain the absolute emissions fluxes at the times specified. These emissions fluxes are then converted into ppm of NO_x emitted into the air mass per minute based on the current value of the inversion height as discussed in the previous section. Note that the exact same emissions rates would be calculated regardless of the relative

values given in the entries for 0900 and 1500, provided only that they are the same and are nonzero. Note also that if all entries in an emissions schedule are zero, then a divide-by-zero error would occur when the CALC command is given for the scenario.

It is important to recognize that the starting and ending time of the simulation (the ST and the ET scenario parameters) are not used when calculating the area under the relative emission flux vs. time curve in the process of determining the absolute emissions fluxes from the total pollution level and the IF parameters. Thus, as indicated in Section IV.D.4 in general the first and last entry in the emissions schedule must either correspond exactly to the start and stop of the simulation, or be zero and within the time range of the simulation. In particular, if the schedule is such that nonzero emissions are specified for a period outside the simulation time, then the total amount of pollutants input into the air mass during the simulation would be less than that indicated by the total pollution level (NOX, BOL, or TML) parameter, by the amount that would have been calculated to be input during the period of time included in the schedule but not in the simulation. This may be appropriate in some applications.

The derivation of absolute emissions fluxes and input rates is exactly analogous for ROG emissions, for emissions with stepped schedules, and for scenarios where PLU=PPM. Note especially that regardless of the value of PLU, the emissions schedules always refer to schedules of emissions fluxes, and not to schedules of rates of change of pollutant concentrations due to emissions. In in scenarios with varying inversion heights, these schedules would be different even when considered in the relative sense.

6. Entering Varied Parameter Values

As discussed in Section IV.A.5, the user has the option of declaring certain types of parameters as variable in order to allow a series of calculations to be carried out with differing values of the parameter as a result of a single CALC command. A parameter can be designated as varied by entering its name as the value to the VAR scenario

option. For example, to indicate that the total NO_x emissions level is to be a varied parameter, with the calculations being carried out with a series of values for this parameter, the command

VAR=NOX

is entered. Parameters which can be varied are LF, IH, RH, ALO and NOX. Note that a parameter which is varied cannot be scheduled, nor can the value for a parameter be specified by giving the parameter name and its value, as would otherwise be the case. Instead, the values for the varied parameter are stored as a list of values, which are defined or modified using the VP or the VPn commands, as discussed below.

The values for the varied parameter can be either numeric, if the varied parameter is LF, IH, RH or NOX, or is the name of a composition file if the varied parameter is ALO. The current list of values for the varied parameter will be displayed by entering the

LIST VP

command. The command

VP,value

will add the new varied parameter value at the end of currently defined list values. For example, if there are no varied parameter values, this command will enter the first value, if there are three values already defined, this will define the fourth value. The order in which they are entered determines the order in which the calculations are carried out as a result of the CALC command. Note that there is no checking the order for numeric varied parameters.

The command

VPn,value

is used to replace the nth current varied parameter value. For example if there were three varied parameter values, VP2,.2 would set the second value to .2, replacing whatever the old value was. If n is greater than the number of varied parameters, the VPn command has the same effect as the VP command, i.e., it will add a new value to the end of the list. However, in this case, the program would inform the user of the parameter number which he actually entered, to indicate that there was no nth varied parameter.

The command

VPn,X

is used to delete the nth varied parameter. Note that in this case the n+1 value then becomes the nth value and so on. Thus, for example, the repeated entry of the VP4,X command will result in the deletion of all but the first three of the varied parameter values. If n is greater than the number of varied parameters, a warning message is given, and the varied parameter values remain the same. The command

VAR,X

will delete all varied parameter values in addition to indicating that no parameters are varied.

7. Nonstandard Parameters

The nonstandard parameter (NSP) option allows the user to include input into the Integration program input files created by the CALC command which are otherwise not supported by the User Interface. This includes, for example, Integration program commands to change values of rate constants or of variable coefficients in the chemical mechanism, or explicitly specifying initial species concentration. This option is not generally required for normal usage, but is included primarily in order to allow the user to change reaction rate constants or variable coefficients in the chemical mechanisms. The nonstandard inputs are included in the

integration input file after all of the input created as a result of the other scenario options, with the exception of the input specifying the varied parameter, thus, it can be used to supersede input created as a result of other scenario options, other than the varied parameter values. It is not necessary for any nonstandard parameters to be input in order for the scenario to be "complete."

The nonstandard parameters consist of a list of entries consisting of a parameter name with up to eight characters, and optionally a numerical value associated with that parameter. The program does no checking as to whether the parameter name will be recognized by the Integration program, or whether the value associated with it is appropriate for that type of parameter, so it is up to the user to assure that legal Integration program input is included. Improper input will result in errors at integration time and usually cause the calculation to fail. The nonstandard parameter is entered into the Integration program input as the unmodified parameter name itself if entered with no value, or as the parameter name, one or more spaces, and the value, if it is given with a numerical value.

The LIST NSP command will give the list of the nonstandard parameter names and values currently associated with the scenario. To add a new nonstandard parameter, enter

```
NSP,name
```

or

```
NSP,name,value
```

To edit an existing nonstandard parameter, or nonstandard parameter value, enter

```
NSPn,name
```

or

```
NSPn,name,value
```

where n is the parameter number, which is shown when the LIST NSP command is given. This replaces both the name and value of the nth entry on the

list, and if there was a value with the old entry and none specified on the NSPn command, the old value is deleted. If n is greater than the number of entries, then the NSPn command is the same as NSP command. To delete the nth nonstandard parameter, enter

NSPn,X

this results in the n+1th entry (if any) becoming the nth entry, etc. Note that this is exactly analogous to the ways in which the VP parameters are created and edited.

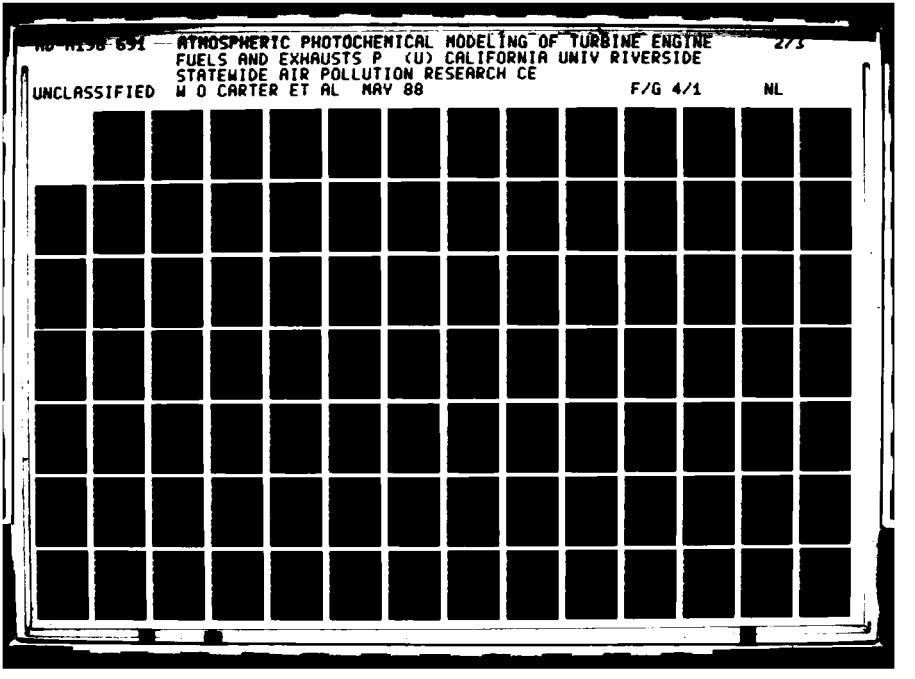
An example of the use of the nonstandard parameter option is to change a reaction rate constant in the mechanism. A reaction rate constant is modified by the integration input command

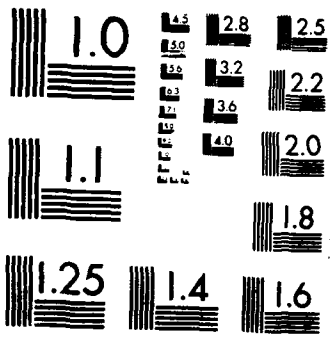
K(rxnlbl) newrateconstant

where "rxnlbl" is the 1-4 character label for the reaction, and "newrateconstant" is the new rate constant, in ppm and minute units (i.e., in units of ppm min^{-1} for zero order reactions, min^{-1} for first order, ppm^{-1} min^{-1} for second-order reactions, etc.). The reaction label for a reaction is given in the reaction listing in the Preparation program output files, and also in the integration program listing output file if the OUT and the WRM output options are selected. The rate constant for any reaction which is not a photolysis reaction or which is not forced to have the same rate constant as some other reaction can be changed in this way. For example, in the chemical mechanisms incorporated in this model, a reaction is included to represent the possibility of first order loss of ozone on surfaces (reaction label = "O3W"), but the default rate constant is zero. To change this rate constant to, for example, 3.33 by 10^{-4} min^{-1} (i.e., 2 percent ozone loss per hour), give the SCEN mode command.

NSP,K(O3W),3.33E-4

This adds a nonstandard parameter named K(O3W) with a value of 3.33 by 10^{-4}, which in turn results in the command


AD-A198 691 ATMOSPHERIC PHOTOCHEMICAL MODELING OF TURBINE ENGINE 2/3
FUELS AND EXHAUSTS P (U) CALIFORNIA UNIV RIVERSIDE
STATEWIDE AIR POLLUTION RESEARCH CE
UNCLASSIFIED W O CARTER ET AL MAY 88 F/G 4/1 NL


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
6.3
7.1

K(O3W) 3.330E-04

being included in the integration input file when a CALC command is given for this scenario.

Note that if rate constants or mechanistic parameters are changed without regard to results of basic laboratory studies, or without testing the model against appropriate environmental chamber data, the validity of the model calculations may be compromised. A full discussion of the capabilities of the Integration program which can be accessed using non-standard parameters is beyond the scope of this document.

8. Output Options

In addition to parameters specifying the conditions of the scenario being calculated, there are additional parameters which determine what types of output are produced as a result of the CALC command. These are of two types: those which determine what output, if any, is included in the integration output listing file, and those which determine whether the results of the calculation are to be plotted or tabulated by the plotting program, and if so which species are to be plotted or tabulated. These options have no other effect on the results of the calculation.

The Integration program listing output file is discussed in Section IV.A.10, above. The OUT option is used to determine whether such a file is desired; if OUT=Y is specified, one is produced; if OUT=N, then no such file is produced unless an error occurred at integration time. If OUT=Y, the output file contains at a minimum a listing of the parameters input, and the initial and calculated concentration time data, as well as giving the calculation label, the calculation number, and other documentation. Additional output can be produced by giving the WDY, WRA, and WPM integration output options values of "Y." If WDY=Y, then the rates of change of the active and the buildup species are included along with the concentration data for each output time. If WRA=Y, the output file gives the rate of each of the reactions at each time the concentration data are output, where the various reactions are identified by their

label. If WRM=Y, the output file also contains a listing of the species, reactions, and variable coefficient values in the chemical mechanism. This "model" listing depends only on the chemical or lumping model used (i.e., the value of the MOD scenario parameter, and not on any of the other scenario parameters. WDY=N, WRA=N, and WPM=N results in this additional output not being included in the integration output file.

If the output options PLT=Y or TAB=Y are selected, then the CALC command produces additional command and input files so that the plotting program is run to produce a plot output file containing plots or tabulations of selected results of the calculation. Plot output files are discussed in Section IV.A.10. If PLT=Y, then plots are produced, if TAB=Y, then tabulations are produced; and if both PLT=Y and TAB=Y, then both plots and tabulations are produced. The species which are plotted and/or tabulated are specified using the PS (for "plot species") command, as described below. Note that the TAB option and the plot species (PS) list is ignored if PLT=N is specified.

The commands used to add or edit species on the plot species list operate in a manner entirely analogous to those used to add or edit the varied or the nonstandard parameter list. A listing of the current plot species is obtained by giving the LIST OUTP command. (This command gives the values of the other output options as well.) To add a plot species to the list, enter

```
PS,name
```

where "name" is any legal model species name (see below and Section III.B). To change the nth entry in the plot species list, enter

```
PSn,name
```

To delete the nth species from the list, enter

```
PSn,X
```

The order in which the species are plotted or tabulated in the plot output file is determined by the order they are entered on the list.

Note that, as discussed in Section III.B, only those species which are included explicitly in the chemical model should be included among the plot species. These are not the same as the ROG species or classes which can be included in composition files, since in most cases these species are represented by model species with different names, where in many cases a given model species being used to represent more than one ROG species. Since only the model species are calculated and saved by the integration program, these are the only species which can be found by the plotting program when plotting or tabulating the results of the calculations. If a species name included in the PS option is not in the model, and PLT=Y specified, the plot output file will contain a message saying that species was not found, but will include plots of other species listed, provided they are in the model. If TAB=Y is specified and an unrecognized name is included, then the table heading will not correspond to the species being tabulated.

As discussed in Section III.A, the model species used to represent ROG species in general depend on which lumping model is used, though the species used to represent inorganics or nonemitted organic products generally the same regardless of the lumping technique. Lists of the names of the legal model species for the various lumping models in the current implementation of the mechanism are given in Tables 3 and 4.

9. SAVE Command

The SAVE command results in the scenario in memory being stored on disk for future use, and so it can be referenced by name by the CALC command. This is done by creating a saved scenario parameter file for the scenario (named name.SCP of VAX systems, or SDname on Cyber systems), and by adding the name of this scenario to the list of available saved scenarios, if it is not already listed there. The list of available saved scenario names is used by the User Interface to determine whether a scenario exists when a SCEN or a CALC command naming it is processed, and is saved on a file named ALLSCEN.LIS on VAX systems, and ALLSCEN on Cyber systems. When a scenario is saved, the program will give a message

indicating whether the scenario is complete, and will also give the scenario name. Note that it is not necessary for a scenario to be complete to be saved, though an incomplete scenario cannot be used in a CALC command. After the scenario is saved, the program returns to the MAIN mode with the scenario still in memory.

If a scenario is not saved, any changes made to it during this session will be lost if one of the following occurs: (a) the user leaves the User Interface program; (b) another scenario is read into memory using the SCEN scenarioname command; (c) a NEW or DELETE command is issued in the SCEN mode (see below); or (d) a CALC command is issued explicitly giving a scenario name, even if the scenario name is the same as the one currently in memory (see Section IV.C.8). Note in particular, if this scenario is named in a CALC command, the version of the scenario which was last SAVED is used, not the version that is currently in memory.

The SAVE command can optionally contain a name for the scenario. If a name is given, the scenario is saved with the name specified, and also the name of the scenario in memory is changed to the new name.

The SAVE command is not the only command which can be used to save a scenario. The scenario will also be saved if the user gives appropriate responses to the prompts resulting from the EXIT or the NEW commands, as discussed below.

10. EXIT and QUIT Commands

The EXIT and QUIT (or Q) commands are used to exit the SCEN mode and return to the MAIN mode. When either command is executed, the scenario remains in memory, and subsequent entry of a SCEN command without a scenario name will return the program to the SCEN mode with the scenario being unchanged, provided that a CALC or SCEN command naming a scenario was not given in the meantime. The only difference between these two commands is that when the QUIT command is given, the MAIN mode is entered immediately, while if an EXIT command is given, the program first asks the user if he wishes to save the scenario. If the answer is "Y," then the scenario is saved before the MAIN mode is entered.

11. NEW Command

The **NEW** command is used to delete the scenario currently in memory and to cause the program to ask for a new scenario name to edit. Before it deletes the current scenario, it asks if the user wants to save it. If the answer is "Y" then the scenario is saved. In either case, the program then asks the user if he wants to see a list of available scenarios, and then asks for a a new scenario name to edit. The behavior of the program at this point is the same as if the user entered the SCEN command from the main mode with when no scenario is in memory. If the user enters "Q" or "QUIT" when the program asks for a new scenario name, the program returns to MAIN mode with no scenario in memory. This command is primarily useful when a series of scenarios are being edited, one after the other.

12. DELETE Command

The DELETE command is used to remove the current scenario from memory and to remove it from the list of scenarios which are recognized by the User Interface as being saved. Thus the execution of this command is essentially equivalent to permanently deleting the scenario, though it is possible to recover it, as discussed below. Only the scenario currently in memory can be DELETEd. In order to check to make sure this is what the user really wants to do, the program then prompts the user for the scenario name, for verification. The user should input the current scenario name. If the name entered does not match, the command is canceled. If the name given matches the name of the scenario currently in memory, the scenario is deleted from the list of available scenarios, and the program returns to the MAIN mode with no scenario in memory.

Note that the DELETE command deletes the scenario only from the list of saved scenarios used by the User Interface to determine if a scenario exists, but does not delete the saved scenario parameter file itself. This file, which is named name.SCP on VAX systems, or SDname of Cyber systems, where "name" is the scenario name, must de deleted explicitly using the appropriate system commands if it is no longer

needed. The user can restore the availability of a DELETEd scenario to the User Interface by manually editing the file containing the list of scenarios recognized by the User Interface, named ALLSCEN.LIS on VAX systems or ALLSCEN on Cyber systems, and adding the scenario name back in. Note that if a scenario name is entered into the list for which no saved scenario parameter file exists, then a "file-not-found" error will occur when attempting to load it into the User Interface through the SCEN or the CALC command.

E. COMP MODE COMMANDS AND OPERATIONS

The COMP mode is used to create, modify, and save composition files. As discussed in Section IV.A.3, composition files consist of list of either (1) emitted ROG species and their relative amounts, or (2) background pollution species and their absolute concentrations in ppm. Files of the first type, referred to as "ROG emissions files" in the subsequent discussion, can be referenced as values for the BOC or the TMC scenario parameters, while composition files of the second type, referred to as "background pollution files," can be referenced as values to the BOC or the ALO scenario parameters. Regardless of which type of files are being processed, the COMP mode commands employed are the same, and the program does not distinguish between when the COMP mode is being used. It is up to the user to assure that the appropriate type of input is included in the different types of composition files, as discussed in Sections IV.A.3 and IV.D.5.a, and in the relevant sections below.

The COMP mode is accessed through the MAIN mode using the COMP command. If the COMP command is entered without a composition file name, the User Interface will prompt the user for a file name or for a question mark to list the currently available composition file names. The user must give a composition file name in order to access the COMP mode. If the user gives the name of a composition file which has been previously created and saved, then the composition data in the named file is loaded into memory for the user to edit. If the name is not found in the list of available composition files (i.e., has not been previously created and saved through the User Interface), the program assumes it is a new

composition file, and the COMP mode is entered with no composition data in memory. Note that the COMP mode operations are independent of whether or not there is a scenario in memory, and they have no effect on the parameters in that scenario.

The names of composition files can consist of 1-5 alphanumeric characters (1-8 characters on DEC systems). However, the names of composition files cannot be the same as legal ROG species name (i.e., a name listed in the first column in Table 2 of Section III.B for the current version of the model), since those names are reserved for referencing the pure compounds when used as values of the BOC scenario parameter (see Section IV.D.5.a). When saved, the data in composition files are stored in files named name.CMP on VAX systems or CPname on Cyber systems, where "name" is the composition file name used by the User Interface. In addition, the User Interface maintains a list of the names of the available saved composition files, which is stored on the data set named ALLCOMP.LIS on VAX systems or ALLCOMP on Cyber systems. Composition files not named in that data sets are not recognized by the User Interface.

A summary of the COMP mode commands is given in Table 10. These commands, and the ways in which species level entries in the composition files are added or modified, are discussed below.

1. Adding or Editing Species and Amount Entries

Composition files consist of entries of species names and numerical values giving their relative or absolute concentrations. If the composition file is used to specify background of aloft pollutant levels, the numerical value given is its concentration in PPM. If the composition file is used to specify ROG emissions (whether base case or test mixture), the value given is its relative amount (on a molar basis) in the mixture. To add or edit a species name and amount entry in the file, enter

```
name=amount
```

or

```
name,amount
```

or

```
name amount
```

TABLE 10. SUMMARY OF COMP MODE COMMANDS.

Command	Explanation
name number	Gives the concentration or relative amount of the named species to include on the composition file. If the species named is already on the the file, its amount is changed to the new value given. If it is not on the file, it is added to the file.
+name number	Increase the concentration or relative amount of the names species by the amount entered. If the species is not on the list, it is added, along with the amount entered.
name,X	Deletes the named species from the current composition file.
ANNOT	Gives a full listing of species which can be included in composition files, along with brief descriptions of the compounds they are used to represent and (for emitted ROG species) the number of carbons used when the NORM command is executed.
DELETE	Deletes current composition file from memory and (if saved) from disk and the list of available composition files. Returns to MAIN mode.
FIND string	Searches list of names of all species which can be included in composition files and lists names with partial match for specified character string.
HELP	Gives COMP mode help display.
HELP category	Gives information about the category specified.
HELP HELP	List all categories on which HELP is available.
LIST	List names and levels of species in current composition file.
NORM	Normalizes the species levels in the current file to give a total of 1 carbon. Used only for composition files to be used for emitted ROG species.
NEW	Deletes current composition file in memory and then asks for new composition file to edit. First asks if user wants to save existing file.
QUIT	Exits COMP mode and returns to MAIN. First asks if user wants to save existing composition file.

TABLE 10. SUMMARY OF COMP MODE COMMANDS (CONCLUDED).

Command	Explanation
SAVE	Saves current composition file in memory. Remains in COMP mode with current file.
SPEC	List names of all species which can be included in composition files. Similar to the ANNOT command, except that only species names are given.

where "name" must be a species name recognized by the User Interface as a legal name to use in composition files. If the species named is not on the composition file in memory, the new entry is added to the list. If the species named has already been entered, its amount is changed to the new value entered. If the species name is not recognized as a legal name to use in composition files, an error message is given, and the composition file in memory is not modified. If a species name is entered without a numerical amount, the user is then prompted to give an amount.

If a "+" character is given before the species name (without any intervening spaces), and if the species is already entered on the file, then the species value is increased by the amount indicated. If the species named is not on the file (and it is a legal species name), then it is added to the file with the value given, i.e., the effect is the same as giving the name by itself. For example, the sequence of commands

```
N-C5 2.0
+N-C4 3.0
+N-C5 3.0
```

results in a composition file with N-C4 with a value of 3.0 (assuming it had not been previously entered on the file), and N-C5 with a value of 5.0.

To delete a species and amount entry from the file, give the command

```
name=X
```

or

```
name,X
```

or

```
name X
```

or give the species name without an amount, and enter "X" in response to the prompt for the amount. An error message is given if the species named is not on the file.

The LIST command is used to give a display of the current species included in the composition files and their associated values. The species are listed in the order in which they were entered. The List command also shows the name of the current composition file.

2. Legal Species Names and the SPEC, ANNOT, and LIST Commands

The SPEC and ANNOT commands are used to give the user the list of the legal species names which can be included in a composition file, and the carbon number associated with each. As discussed in Section IV.E.3 below, the carbon number refers to the number of carbons in the emitted ROG species used when the NORM command is executed, and is zero for inorganic and nonemitted organic product species. These data are taken from the file named SPECIES.LIS on VAX systems or SPECIES on Cyber systems. The difference between these two commands is that the ANNOT command gives a full listing of the SPECIES file, including descriptive comments, while the SPEC command lists only the species name and carbon number, and thus gives a significantly briefer display. The user can edit the SPECIES file to modify or add the descriptive comments, but he should not add or delete species unless the corresponding changes have also been made in the chemical models. In the current implementation of the model, the species which can be included in composition files are those listed in Tables 2 and 3 in Section III.B, and these are the names which are displayed as a result of the SPEC or ANNOT commands.

Although more brief than the output caused by the ANNOT command, the SPEC command still produces a relatively large amount of output, which more than fills the screen of a CRT terminal. Thus, in order to obtain a listing of a subset of species, the command

FIND substring

where "substring" is any set of 1-8 alphanumeric characters, can be used. This command will result in a display of all species which can be included in composition files whose names contain the substring indicated. If no substring is given, the program prompts for one. For example, in the initial implementation of the model, the command

FIND C8

will result in the display

```
N-C8      ( 8.0)
ISO-C8    ( 8.0)
BR-C8     ( 8.0)
CYC-C8    ( 8.0)
C8-OLE1   ( 8.0)
C8-OLE2   ( 8.0)
```

This can be useful if the user forgets the exact spelling of a species name.

Note that as discussed in detail in Section III.B and Section IV.D.5.a, there are two types of species which can be entered in composition files: those which can be used in files representing emitted ROG species and those which should be used only in files representing background or aloft pollutants. However, the program does not know how the composition file being created is going to be used, so it is up to the user to assure that files to be used to specify base case or test mixture ROG emissions contain only names for emitted ROG species. In the current implementation of the model, the first type of species, the emitted ROG species, are listed in Table 2 in Section III, while the second type, the

inorganic and nonemitted organic product species are listed in Table 3. The output resulting from the ANNOT command will identify the type of species when they are listed. In addition, only those species which can be used as emitted ROG species have nonzero carbon numbers associated with them, since (as discussed in Section IV.E.3 below) the carbon number is only used for the NORM command, and the NORM command is only appropriate for composition files to be used for ROG emissions. (Thus, for example, the nonemitted organic product species are given carbon numbers of zero even though they in fact have carbons.) The carbon numbers for the species are included in the ANNOT, SPEC, and FIND displays.

Note that the output produced by the SPEC, ANNOT and FIND commands are independent of which species are included in the current composition file.

3. NORM Command

The NORM command is used to normalize composition files for ROG emissions such that the product of the number of carbons and the relative amounts for each of the species in the file sums up to one. As discussed in Section IV.D.5, composition files for ROG emissions should be normalized in this way so that the SCEN parameters giving the total levels of emitted ROG pollutants (i.e., BOL and TML) are in the proper units. The number of carbons for each ROG species is obtained from the SPECIES file as discussed above, and is included in the output produced by the SPEC, ANNOT, or FIND commands. For example, if one wished to create a normalized composition file for a test mixture containing propene, n-butane and toluene (which have respectively 3, 4, and 7 carbons) such that the molar amounts of propene and toluene are equal, and there is twice as much n-butane as either of the other two, the following commands can be entered.

```
PROPENE,1.0
TOLUENE,1.0
N-C4,2.0
NORM
```

The resulting contents of the composition file, as indicated by the output of a LIST command, would then be:

```
PROPENE    5.5556E-02
TOLUENE    5.5556E-02
N-C4       1.1111E-01
```

Note that the total number of carbons in the mixture in this composition file sums up to one.

If the mixture being represented in the composition file contains chemical compounds which have more (or fewer) carbons than assigned to the ROG species used to represent it in the model, the special model species called INERT can be used to represent the additional carbons in the composition file for normalization purposes. The INERT species has one carbon, but including it in a composition file has no effect on the results of a model calculation. For example, propylbenzene (with 9 carbons) is represented in the model by ALK1BENZ, which (since it is represented by toluene in the model) has only seven carbons. Thus, to represent all the carbons in propylbenzene in composition files for normalization purposes, one should include two INERTs for each ALK1BENZ being used to represent propylbenzene. Thus, if one wanted to create a normalized composition file for a 1:2:1 (molar) mixture similar to that given in the previous example, except that instead of toluene it contains propylbenzene, one could enter:

```
PROPENE,1.0
ALK1BENZ,1.0
+INERT,2.0
N-C4,2.0
NORM
```

(The "+" form for entering INERT, though unnecessary in this example, is recommended in case it is being used to represent otherwise unrepresented carbons in outer species as well.) The resulting contents of this composition, as indicated by a LIST display, would then be:

```
PROPENE    5.0000E-02
ALK1BENZ   5.0000E-02
INERT      1.0000E-01
N-C4       1.0000E-01
```

As indicated above, the NORM command is only appropriate for composition files which are intended to be used to represent ROG emissions, and not for files representing background or aloft pollutants. Since inorganic and nonemitted organic species, should only be included in files for background or aloft pollutants, this implies that the NORM command should not be used when editing files which contain such species. Therefore, if a NORM command is given in a file containing such species, an error message is given and the file is unmodified. The error message gives the name of the first such species encountered in the file; note that if there is more than one nonemitted species only the first one on the list is given. The program recognizes such species because inorganic and nonemitted organic product species are assigned carbon numbers of zero.

4. SAVE Command

The SAVE command will result in the composition file in memory being saved on disk and its name added to the list of the saved composition files available to the User Interface program. The composition file data are saved in files named name.CMP on VAX systems and CFname on Cyber systems, and the list of available composition file names used by the User Interface is named ALLCOMP.LIS or ALLCOMP for the two respective systems. The SAVE command does not remove the composition file from memory, and the program remains in the COMP mode after it is executed.

The composition file can also be saved as a result of the QUIT or the new commands, if the user gives the appropriate response to the prompt resulting from that command, as indicated below.

5. QUIT and NEW Commands

The QUIT command is used to return to the MAIN mode. Before returning to the MAIN mode, the program asks if the user wants to save the scenario, and if a "Y" response is entered at that prompt, the scenario is saved. (The default is for the scenario not to be saved, as indicated by the prompt.) Unlike the SCEN mode, after the COMP mode is exited the composition data no longer remain in memory, and these data are lost unless they are saved before reentering the MAIN mode.

The NEW command operates like the QUIT command, except that instead of returning to the MAIN mode, the program asks for a new scenario name to edit. This command would be used if a series of different composition files are being created, one after the other.

6. DELETE Command

The DELETE command is used to delete a previously saved composition file and to remove the file from the list of saved composition files used by the User Interface. It can only be used to delete the file currently in memory, and only then if it has been previously saved. If the file currently in memory has not been saved (i.e., if the file name is not found on the list of saved composition files available to the User Interface, the program gives a message indicating that the file has not been saved, and nothing is done. The program still remains in SCEN mode with the current composition data in memory. If the file has been previously saved, the program asks the user to enter the name of file being deleted, for verification. If the name does not match the name of the current composition file, nothing is done, and the program informs the user of this fact. If the names match, the file is deleted and its name is removed from the list of available saved composition files, and the program returns to the MAIN mode.

F. PLOT MODE AND PLOTTING PROGRAM COMMANDS AND OPERATIONS

The PLOT mode is used to give access to those features of the Plotting program which are normally needed when using this model for airshed calculations. By accessing the PLOT mode of the User Interface, or by running the Plotting program directly from system level, one can obtain plots or tabulations of concentration-time data resulting from the model calculations, obtain plots of data from two different calculations together for easy visual comparison of effects of changes in model input, save plots or tabulations on data files for subsequent printing or examination, and create plots or tabulations in batch mode by accepting command input from files rather than from the user terminal. Since the commands and options of the PLOT mode of the User Interface are a subset of those of the Plotting program itself, the discussion in this section applies equally well to both programs. The only difference is that the full Plotting program has several other features and options not normally useful to the user of this model for airshed calculations, and which are therefore not discussed in this document, and that when exiting the Plotting program one returns to system level, while when exiting the PLOT mode of the User Interface, one returns to the MAIN mode of the User Interface. Accessing the PLOT mode of the User Interface has no effect on the scenario in memory or the status of any created or submitted calculations.

The Plotting program or the PLOT mode of the User Interface has three main submodes or levels of commands, which one enters and exits by entering carriage returns (<cr>'s) at the prompts. These are designated by OPTION, CURVE (or DATA), and ENTER levels, as indicated by the prompts that they give. The OPTION level is the first one encountered when invoking the Plotting program or entering the PLOT mode of the user interface. Commands entered in this mode control the general operation of the program, such as, for example, whether plots or tabulations are to be produced, or whether output goes to the terminal or a file. The CURVE, or DATA, level is used to tell the program which species to plot or tabulate. The prompt CURVE is given if the data are to be plotted, while the prompt DATA is used when the data are to be tabulated, and the operation of the

program in this level is also somewhat different, depending on whether the data are to be plotted or tabulated. After the names of the species to be plotted or tabulated are entered, the program gives an ENTER prompt requesting the calculation number(s) whose data are to be plotted or tabulated. A "QUIT" or "Q" command can be entered from any of these modes to terminate the plot program or to exit the PLOT mode of the User Interface and return to the MAIN mode.

1. Plot Options

The Plotting program recognizes a number of options which control the operation of the program. These options are specified by giving the appropriate commands when the program is in the OPTION level, i.e., when it displays the "PLOT OPTION:" prompt. The program is in this level initially, and it can be reentered by entering <cr> at the "CURVE 1:" CURVE level prompt, or by entering <cr> at the DATA level prompt. A summary of the available commands and options which can be entered at this level are given in Table 11. The user can enter HELP at the OPTION prompt to obtain a list of the options available to him.

As shown in the table, there is a default set of plot options which are always in effect when the Plotting program is initially run or whenever the PLOT mode of the User Interface is entered. If the default options are suitable, the user needs not enter any options. If the user changes to a non-default option, that option remains in effect until it is canceled, or until the PLOT mode or plotting program is exited.

a. TABL and NOTABL Options

The TABL and NOTABL options control whether plots or tabulations are to be produced. The default mode when the Plotting program is run, or when the PLOT mode of the User Interface is entered the first time, is to produce plots rather than tabulations. The TABL command can be given to have tabulations produced rather than plots. This remains in effect until it is canceled, or until the PLOT mode or the Plotting program is exited. The NOTABL cancels the TABL command, and returns the

TABLE 11. SUMMARY OF PLOT OPTION COMMANDS.

Command	Explanation
<cr>	Exits the OPTION level and enters the CURVE or DATA level, depending on whether the curve or tabulation option was selected.
IN=file	Takes the Plot command input from a file. The input comes from the file called "file.PLT" on VAX systems and from "PLTfile" on Cybers, where "file" is the name specified on the IN command. After the commands are executed, the program exits, or returns to the MAIN mode if run via the User Interface.
OUT=file	The plot output goes to a file rather than the user terminal. The file produced is called "file.PLO" on VAX systems, and PLOfile on Cybers.
OUT=CRT	Cancels the effect of the OUT=file command and causes any subsequent command to go to the terminal. This is the default mode.
PPP=n	Determines how many plots go on a page if the output is going to a file. This command has no effect if the output is going to the user terminal, or if tabulations are being produced.
TABL	Indicates that tabulations are to be produced rather than plots. The default mode is for plots to be produced.
NOTA	Cancels the "TABL" option and indicates that plots are to be produced. This is the default mode.
CLOCK	Causes the X-axis (time) label on the plots to be given as clock time (HHMM) rather than as minutes after midnight. The default is for the time axis to be labeled as minutes after midnight, since this is how the data are stored. Cancels the HOUR option if it is in effect.
HELP (or HE)	Gives a display of the available plot options.
HOUR	Causes the X-axis (time) label on the plots to be given as hours after midnight, rather than minutes after midnight, which is the default. Cancels the CLOCK option if it is in effect.

TABLE 11. SUMMARY OF PLOT OPTION COMMANDS (CONCLUDED).

Command	Explanation
MINU	Cancels the CLOCK or the HOUR option, and causes the X-axis (time) label to be given as minutes after midnight, which is the default.
XMAX=minutes	Determines the maximum time for which data are plotted or tabulated. Must be given as minutes after midnight.
XMIN=minutes	Determines the minimum time for which data are plotted or tabulated. Must be given as minutes after midnight.
NOXMAX	Cancels both the XMAX and the XMIN option. Same as NOXMIN. Results in no limit being placed on the time values for the data being plotted. This is the default mode.
NOXMIN	Same as NOXMAX.
Q (or QUIT)	Return to MAIN mode. Close plot output file (if any).

mode where plots are produced rather than tabulations. If the NOTABL or the TABL commands are given when the options they select are already in force, they have no effect and no error message is given.

If the program is in the default mode and plots are to be produced, a <cr> at the "PLOT OPTION:" prompt will result in the prompt "CURVE 1: ," which requests for input specifying the types of data to be plotted on the first curve of the plot. If the TABL option is in effect, a <cr> at the "PLOT OPTION:" prompt will result in the prompt "DATA: ," which requests for input giving the types of data to be tabulated. The input used to specify data to be plotted or tabulated are discussed in Section IV.F.2 and IV.F.3, below.

b. CLOCK, HOUR, and MINU Plot Options

The plots produced when the default, or NOTAble, option is in effect consists of concentration data given on the vertical (Y) axis,

plotted against time on the horizontal, or X, axis. The CLOCK, HOUR, and MINUtes options determine how the time label on the X axis is to be specified. The time data used by the Integration program during the normal usage is always in minutes, and when used in airshed calculations (such as when run through the User Interface) they are always given as minutes after midnight. Thus the default display for the X, or time, axis label is to give it in minutes after midnight. If this is not desired, the user can select the CLOCK option to have the time labeled as clock time (HHMM), or the HOUR option to have it labeled as hours rather than minutes. The MINUtes option returns the program to the default state of outputting the time axis label in minutes. For example, in a scenario with start and stop times of the simulation of 0800 and 2100 hours, respectively, the numbers given to label the time axis are (480, 600, ..., 1260) when the default (MINUtes) option is in effect, are (8, 10, ..., 21) with the HOUR option, and are (800, 1000, ..., 2100) with the CLOCK option.

These options have no effect if the TABL option is selected; the time data are always tabulated as they are read from the Integration program output, which (for calculations created by the User Interface) are minutes after midnight. If the user returns to the MAIN mode, then reenters the PLOT mode, the default MINUtes option is in effect, regardless of what had been specified previously.

c. XMAX, XMIN, NOXMAX, and NOXMIN Options

The default mode for the Plotting program is to plot or tabulate all the concentration-time data produced and saved by the Integration program. If plots or tabulations of only a subset of the available data are desired, then the XMAX or XMIN options can be used to specify, respectively the minimum or the maximum time time for which the data are output. One or both of these options can be specified. The formats for these these commands are

XMAX=minutes

and

XMIN=minutes

where "minutes" is the minimum or maximum time value, in minutes (or minutes after midnight for airshed calculations) for which the data are to be output. Note that the values for XMAX or XMIN must always be in minutes after midnight for airshed calculations, regardless of whether the CLOCK or HOUR option is in effect. Note also that if no points are saved on the calculation for the time interval between XMIN and XMAX, such as would occur if XMAX $\leq$ XMIN, or if XMAX is less than the initial time of the simulation or XMIN is greater than the final time, then the program will give the error message "NO DATA POINTS IN CALCxxxx.CDT FILE" at the time the calculation run number is input.

The NOXMAX or NOXMIN options cancel the effect of limiting the time intervals to be output, and returning the program to the default state. They both have the same effect, i.e., NOXMAX cancels the effect of both XMAX and XMIN, and likewise for NOXMIN. The XMAX and XMIN options are also canceled if the user leaves the PLOT mode and reenters it later during the same User Interface session.

d. File Output

The user has the option to send the plot output to a file rather than being displayed at the user terminal. This is done by giving the command

OUT=name

where "name" is a string of 1-8 alphanumeric characters (1-4 on Cyber systems) from which the plot output file name is determined. The name given the output file is "name.PLO" on VAX systems, and "PLOname" on Cybers. When this command is given, all output but the prompts for input go to the file named. To cancel the effect of this command, which closes the PLO file and causes the subsequent output to go to the user terminal,

give the command

OUT=CRT

The output file is also closed when an OUT=filename is given or when the PLOT mode is exited.

Note that if the name of an existing plot output file is given, the existing file is superseded by the new output file. The program does not append new plots to existing files which have been closed as a result of exiting the program or giving an OUT= command to open another file. This means that if OUT=name is given for with the same file name as the one currently open, the file currently opened is closed and a new version of the file with the same name is opened.

If the output going to the file are plots, the default is for two plots to be produced per page. This gives reasonably large plots for printer output. If this is not desired, then the Plots Per Page (PPP) option can be used to determine how many plots will appear per page on the output file were it to be printed. The format of the command giving this option is

PPP=n

where n is an integer between 1 and 4 which specifies the number of plots per page. If a number outside that range is given, an "UNRECOGNIZED OPTION" error message is given. This option has no effect if the TABL option is in effect, nor has it any effect if the output is going to the user terminal.

It should be noted that for displaying plot output files on a CRT terminal, three plots per page is more appropriate than the default of two, since if PPP=3, the resulting plot just fits on a normal CRT screen. Therefore, plot input commands produced by the User Interface program when the PLT=Y option is specified (see Section IV.D.8) always include the option PPP=3, so the user can see the entire plot when the DISPLAY PLOT User Interface command is given.

e. File Input

The plot options and commands can be input from a file, as well as from the user terminal. To utilize this option, the user can use a system editor to create a file called "name.PLT" on VAX systems, or "PLTname" on Cybers, which contain plot input commands and options. The format of those commands are exactly the same as if they were entered at the user terminal, with blank records (or records containing only spaces) being used to switch from one mode, or input level, to the other. In order for the program to take input from that file instead of from the terminal, the command

IN=name

is given, where "name" refers to the PLT filename as indicated above. At this point, all plot commands are taken from the file, and the PLOT prompts are not shown at the terminal (though error messages or warnings are). When the end of file is reached, the Plotting program terminates or (if run from the User Interface) the program returns to the MAIN mode of the User Interface.

Note that if the IN= command is used by itself, the plot output goes to the user terminal just as if the user had entered the commands by itself. Thus, in order to run the program in batch mode, where the input is taken from one file and the resulting output goes to another, the IN= and OUT= commands are used together, with the OUT= command either being the first command in the input file, or the OUT= command being given immediately before the IN= command is given. (The OUT= command has to come first, because after the IN= command is given, no more PLOT input is requested from the user.) An example is the utilization of the Plot program on VAX systems by the User Interface in processing CALC commands for scenarios where the plot output (PLT=Y) option is selected. In this case, it creates the plot input file CALCnnnn.PLT (where nnnn is the calculation number) and a command file giving the plot input commands

```
PPP=3
OUT=CALCnnnn
IN=CALCnnnn
```

which results in an output file called CALCnnnn.PLO being created based on the commands given in CALCnnnn.PLT. (The PPP=3 is an additional option the User Interface adds, which it could have also included in the PLT input file.) The user can give similar commands to create plot output files using file input using file names of his choice.

[When the Plot program is run on Cyber systems, the F=name option on the XPLT command line produces the same net result as the combination of IN= and OUT= options. For example, the command (given at Cyber system level)

```
GET,XPLT
XPLT,F=name
```

has the same effect as the commands

```
GET,XPLT
XPLT
OUT=name
IN=name
```

or

```
GET,RUNSCEN
RUNSCEN
PLOT
OUT=name
IN=name
QUIT
Y
```

where the first two examples involve invoking the Plotting program itself, and the third example involves use of the User Interface (where the last two commands are used to exit from the MAIN mode back to system level).]

The user has the option to include comments in plot input files if desired to document their contents. Comments are any records with a "!" in the first column. Comment records are ignored by the program.

f. QUIT Command

The "QUIT" or "Q" command is used to exit from the plotting program or to return to the MAIN mode of the User Interface if run as the PLOT mode of that program. This command can also be entered in response to the CURVE, DATA or ENTER prompts, and the effect will be the same. If an output file has been opened by the OUT=option, it is closed. All options are reset to their default values. The "QUIT" or "Q" command can be given at any PLOT mode or plotting program prompt; the effect is the same. If plot input is taken from an input (PLT) file, it is not necessary to include this command, since encountering an end-of-file has the same effect. However a "Q" or "QUIT" can be included in such files; the effect would be that the program would ignore any subsequent input. On VAX systems, a "control-Z" has the same effect as a QUIT command, since it simulates an end-of-file.

2. Specification of Species to Plot or Tabulate

After the user has entered all desired options, he enters a blank line, or <cr>, to exit the OPTION level, and enter the CURVE or DATA, level, from which the names of the species whose concentration-time data are to be plotted or tabulated. If plots are to be produced the initial prompt is "CURVE 1: ," while if the TABL option is in effect, the prompt is "DATA: ." The types of input expected at either prompt are essentially the same; the main difference is that if plots are being produced, there can be more than one "curve," or set of data displayed on the same plot (which in general may be from different calculations), whereas if

tabulations are being produced, data from only one calculation can be tabulated at a time. Before discussing these differences, the common features for the input options in the CURVE or DATA modes are discussed.

In response to a "CURVE 1: " or "DATA: " prompt, the data to be plotted can be entered in one of the following formats:

C name1 name2 name3 (etc)

or

char-C name1 name2 (etc)

where "name1," "name2," etc. are the names of the model species to be plotted, and "char" can be any single nonblank printable character. The "C" character is the run type code, which indicated the type of data to be plotted or tabulated. In this application, the only run type code of interest is the "C," which is used to indicate that the data to be plotted or tabulated are calculated data. (Other codes can be used for other applications. For example a "D" instead of a "C" can be entered to plot results of environmental chamber experiments, if such data are available. The other codes are not documented here because they are not required for normal use of this model, and the data sets required to utilize them are not delivered with this model. If other codes were used the format of the rest of the CURVE or DATA input would be the same, except that the names given to the experimental data channels, which may also include specifications of the instrument and chamber side number from which the data was taken, are given instead of the species names used in the model calculations.) The list can contain from 1 to 20 names, each separated from each other, or from the "C" (or other) code character, by a single blank.

If the list of model species to be plotted or tabulated requires more than 80 characters to be input, it is necessary to continue the CURVE or DATA command line. This is done by including an "&" character at any place, and then continuing the command on the next line. The "&" character can go any place in the list, even in the middle of a name

(though that is not recommended), and it is important to note that it does not take the place of a blank. For example, the following four CURVE or DATA command lines to plot "O3" and "NO2" data are equivalent,

```
C O3 NO2

C O3 &
NO2

*-C O3 N&
O2

C O3&
NO2
```

while the line

```
C O3&
NO2
```

is interpreted as "C O3NO2," which would result in an error unless there happened to be a species named "O3NO2" in the model.

The alternative format "char-C name(s)" can be used to control which character is to be used as the plot symbol. The default is to use the symbol "1" for the first curve, "2" for the second, etc. This overrides this default, and the character input is used instead. For example, "*-C O3 NO2" causes the plots to be displayed with the "*" symbol. This format is useful only when specifying data to be plotted; while legal for specifying data to be tabulated, it has no effect on the tabulations output.

Each model species name given on the CURVE or DATA input lines here must match exactly with the name of a species in the chemical model being used in the calculation whose data are being plotted. The considerations involving legal names are exactly the same as those involved in the names of the "Plot Species" (PS) output options of the

scenarios in the User Interface, as discussed in Section IV.D.8. The valid model species names for the chemical models included in the initial distribution of this model are listed in Tables 3 and 4 in Section III.

If the data are being tabulated (i.e., if the TABL option is in effect), then after the "C" code and the species names are listed in response to the "DATA: " prompt, the program then enters the ENTER level and prompts for the calculation number for the calculation whose data are to be displayed. The user can then enter calculation numbers for repeated tabulations of the concentration-time data for the same sets of species for any number of calculations, as discussed below. After all the desired tabulations of this set of species have been produced, the user could enter <cr> at the ENTER prompt, to cause the program to return to the DATA level to accept names for other groups of species to tabulate. If a <cr> is entered at the DATA prompt, the program will return to the OPTION level, where additional options can be entered, or a <cr> can be entered there to return to the DATA or (if the NOTAble command was given to return to plot mode) the CURVE prompts, etc. A "QUIT" or "Q" entry at any point would terminate the Plotting program or the PLOT mode of the User Interface.

If the data are being plotted, the user is given the opportunity to enter additional lists of species to be plotted with the species listed in the previous CURVE entries. Thus, after the list of species for the "CURVE 1: " prompt is given, the program then gives the "CURVE 2: " prompt, at which the user can either (1) enter <cr>, to cause plots of the species listed in response to the "CURVE 1: " prompt to be created, with a separate plot for each species listed, or (2) enter a new list of species to be plotted, in the same format as indicated above. The first species listed on this list will then be plotted on the same plot with the first on the previous list, the second will be plotted with the second, etc -- i.e., the species input here will be the second "curve" of a series of plots with 2 (or more) curves. The program then prompts for "CURVE 3: ," etc., and will continue prompting for additional curves until <cr> is entered at the prompt, or until 12 curves have been defined. The program will then enter the ENTER mode to ask for the numbers for the calculations from which the data for the various curves are to be taken. The user then

can enter as many sets of calculation numbers for which plots of the data specified are desired, and then can either "QUIT" the program, or give a <cr> at the ENTER prompt to return to the "CURVE 1: " prompt. A <cr> at the "CURVE 1: " prompt returns the user to the OPTION mode.

It is not necessary that the same number of species be specified for each curve. If more species are specified for some curves than on others, then the last plots will contain plots only of the species specified for the curves with the greatest number of species.

The specification of the species to be plotted for curves other than the first can be entered as shown above for the format required for the "CURVE 1: " or "DATA" input, though two additional options are available. If the user enters:

```
C "
```

or

```
char-C "
```

then the exact same species as specified on the previous curve are plotted. Again the "C" code is required to indicate that the data to be plotted are calculation results, and the "char"- option can be used to control the character to use on the plots. This is particularly useful when comparing graphically the results of two different calculations, on simulations of the same species.

In addition, on all curves but the first, a blank character can be used in place of the "C" (or any other) code before the species list, in order to have the data from this curve being taken from the same calculation as the data from the previous curve. For example, to plot O3, NO and NO2 on one plot, and C:CC (propene) and TOLU (toluene) together on a second plot where all these data are from the same calculation, one could enter

```
C O3 C:CC
NO TOLU
NO2
```

or alternately (to use "*" for the first plot, "x" for the second, and "o" for the third

```
*-C 03 C:CC
x- NO TOLU
o- NO2
```

Note that in this case, only one calculation number, not three, need be input at the ENTER level to do the plots.

3. Specification of Calculations to Plot or Tabulate

After the names of the species to be plotted or tabulated are specified, the program enters the ENTER mode, where the number(s) for the calculations whose data are to be used are requested. The prompt given is

```
ENTER (nn) RUN NO(S)
```

where nn is the maximum number of calculation numbers which will be accepted. In the case of tabulations (i.e., if the TABL option is in effect), the maximum number of calculations is always 1, since data from only one calculation can be tabulated at a time. In the case of plots, the maximum number of calculations is the number of "curves" which were specified, other than those curves which were designated as having their data being taken from the previous curve (i.e., other than those curves where a blank, rather than a "C," was used as the run type code. All calculation numbers entered for a given set of plots must be given on one line.

In response to this prompt, the user can then enter the calculation numbers, separated by a blank or a comma. As discussed in Section IV.A.7, each calculation is assigned a run number, which can range from 1 to 9999, and the results of the calculations are stored in files named CALCnnnn.CDT on Vax systems, or CDTnnnn on Cybers, where nnnn is the run number. The plotting program then reads the CDT data set indicated by the number input. In the case of tabulations, only one number is entered;

in the case of plots with multiple curves, one number is entered for each curve with a nonblank (i.e., "C") type code, with the first number being the calculation number whose data will create the first curve on each plot, the second being used for the second curve, etc. If fewer than the maximum number of calculation numbers are given, then fewer than the maximum number of curves will be shown on each plot. Two commas can be used to indicate that no run number is given for an intermediate curve, e.g., if three curves are defined, and the user enters

```
100,,101
```

at the "ENTER" prompt, the first curve uses the data from calculation number 100, the third uses the data from calculation 101, and the species named on the second curve are not plotted, since no calculation number for that curve is given.

As discussed in Section IV.A.7, the user can also enter "relative" calculation numbers in response to the "ENTER" prompt. "Relative calculation numbers are numbers which are zero or negative, where zero refers to the last calculation which has been executed by the Integration program, -1 refers to the calculation number which is one less than the last, etc. Note that this is not the same as the last calculation prepared by the User Interface, since some of those calculations have not yet been carried out. The last calculation executed by the Integration program is stored on the file named CALC.PRM on VAX systems, and CALCPRM on Cybers. When a calculation number of zero or less is given, the calculation number used is obtained by adding the "relative" calculation number to the number stored in CALC.PRM. If a positive number is given, the number in CALC.PRM is not used.

The plots or tabulations are produced immediately after the calculation numbers are given. The user will see the plots or tabulations at the terminal, unless the OUT=filename option is used, in which case the output goes to a file instead. After the plots or tabulations have been produced, the program will then prompt again for calculation numbers. The user can then enter additional numbers to produce plots or tabulations of the same species from different calculations, and there is no limit on how

many times this can be done. To exit this mode, the user can either (a) enter a <cr> at the "ENTER" prompt, in which the program enters the "CURVE 1" or "DATA" mode to define a new set of species to be plotted or tabulated, or (b) enter "Q" or "QUIT" to exit the Plotting program, or the PLOT mode of the User Interface, entirely. To get to the "OPTION" mode from the "ENTER" mode, the user must enter <cr> twice.

4. Examples

Several examples of use of the Plotting program and the PLOT mode of the User Interface are given in Table 12. These examples all start with the program being at the OPTION level, with all the options at the default, and end with the Plotting program or PLOT mode of the User Interface being exited. The input in these examples could be entered interactively, or in a file referenced by the IN= option (or in the F= option on the command line for running the Plotting program on Cybers). Note that the prompts and the user (or file) input are shown in separate columns in the table for clarity, and that although all user inputs are terminated by a <cr>, the <cr> notation is only shown if that is the only input on the command line. The output produced by each of these examples is indicated in the table immediately following the commands which cause the output to be produced.

TABLE 12. EXAMPLES OF PLOT MODE INPUT OUTPUT.

Prompt or output produced	User input
Example 1:	
PLOT OPTION	<cr>
CURVE 1:	C O3
CURVE 2:	<cr>
ENTER (1) RUN NO(S)	0
Concentration-time plots of ozone for last calculation executed by the Integration program are displayed at the terminal	
ENTER (1) RUN NO(S)	-1
Concentration-time plots of ozone for second to last calculation executed by the Integration program are displayed at the terminal	
ENTER (1) RUN NO(S)	<cr>
CURVE 1:	C NO2
CURVE 2:	<cr>
ENTER (1) RUN NO(S)	0
Concentration-time plots of NO2 for last calculation executed by the Integration program are displayed at the terminal	
ENTER (1) RUN NO(S)	-1
Concentration-time plots of NO2 for second to last calculation executed by the Integration program are displayed at the terminal	
ENTER (1) RUN NO(S) QUIT	
The system prompt is produced if the command given to the Plotting program; the "MAIN: " prompt appears if the PLOT option of the User Interface is being used.	
Example 2:	
PLOT OPTION:	<cr>
CURVE 1:	C O3 NO NO2 C-PAN
CURVE 2:	C "
CURVE 3:	<cr>
ENTER (2) RUN NO(S)	3120,3123

TABLE 12. EXAMPLES OF PLOT MODE INPUT OUTPUT (CONTINUED).

Prompt or output produced	User input
Four concentration-time plots are produced, each with two curves, with the curve labeled "1" using the data from calculation no. 3120, and the curve labeled "2" using the data from calculation 3123. The first curve contains ozone data, the second contains NO, the third NO2, and the fourth C-PAN, where the names are the names of the species in the model.	
ENTER (2) RUN NO(S)	3134
Four concentration-time plots are produced, each with one curve, labeled "1," using data from calculation number 3134. The first curve contains ozone data, the second NO, etc.	
ENTER (2) RUN NO(S)	<cr>
CURVE 1:	<cr>
PLOT OPTION:	TABL
PLOT OPTION:	<cr>
DATA:	C O3 NO NO2 C-PAN HCHO
ENTER (1) RUN NO(S)	3132
A tabulation of data from calculation 3132 is produced with six columns of data: the first being time (in minutes after midnight), and the other five being calculated concentrations of O3, NO, NO2, C-PAN, and C:C (ethene).	
ENTER (1) RUN NO(S)	3133
The same output as above, except that the data is from calculation 3133.	
ENTER (1) RUN NO(S)	<cr>
DATA:	C O3 NO NO2 C-PAN HCHO C:C C:CC
ENTER (1) RUN NO(S)	3133
Two groups of tabulations are produced. The first consists of the same as above, i.e., give tabulations of O3, NO, NO2, C-PAN, and HCHO, and the second containing three columns, one for time, the other two containing data for C:C (ethene) and C:CC (propene). Since the maximum number of columns which can be output in a tabulation is six, if more than five species are tabulated, more than one set of tabulations will be produced.	

TABLE 12. EXAMPLES OF PLOT MODE INPUT OUTPUT (CONTINUED).

Prompt or output produced	User input
ENTER (1) RUN NO(S)	<cr>
DATA:	<cr>
PLOT OPTION:	NOTA
PLOT OPTION:	<cr>
CURVE 1:	*-C O3 NO NO2 C-PAN HCHO C:C C:CC
CURVE 2:	<cr>
ENTER (1) RUN NO(S)	3133

The species which were tabulated above are not output as plots, with a separate plot for each species, and one curve on each plot, with the curve labeled "*."

ENTER (1) RUN NO(S)	QUIT

Example 3:

PLOT OPTION:	CLOCK
PLOT OPTION:	<cr>
CURVE 1:	C O3 HCHO C-PAN
CURVE 2:	NO2 CCHO
CURVE 3:	NO
CURVE 4:	<cr>
ENTER (1) RUN NO(S)	0

Three plots are produced, all having data from the last calculation executed. The first has plots of O3, NO, and NO2 together, respectively labeled "1," "2," and "3." The second has HCHO and CCHO (formaldehyde and acetaldehyde) plotted together, and the third has C-PAN plotted by itself. The time axes on the plots are labeled with clock time.

ENTER (1) RUN NO(S)	<cr>
CURVE 1:	*-C O3 HCHO C-PAN
CURVE 2:	*- NO2 CCHO
CURVE 3:	*- NO
CURVE 4:	C X X C-PAN
CURVE 5:	<cr>
ENTER (2) RUN NO(S)	1313,1314

The following warning messages are output because "X" is not a name of a species on the model used:

```
CALC 1314: NAME = 'X               ' NOT FOUND
CALC 1314: NAME = 'X               ' NOT FOUND
```

TABLE 12. EXAMPLES OF PLOT MODE INPUT OUTPUT (CONTINUED).

Prompt or output produced	User input
Then three plots are produced. The first two are the same as above, except instead of being labeled "1," "2," or "3," the all the curves are labeled "*." The last contains two curves of calculated data for C-PAN, the curve labeled "*" being from calculation 1313, and the curve labeled "4" being from calculation 1314.	
ENTER (2) RUN NO(S)	<cr>
CURVE 1:	QUIT

Example 4:

In this example, it is assumed that the LPARM model was used for calculation 201, and that the LM2ARM model was used for calculation 202. See Section III for a discussion of these models and the model species used by each.

PLOT OPTION	<cr>
CURVE 1:	C N-C4 N-C5 TOLUENE
CURVE 2:	C "
CURVE 3:	<cr>
ENTER (2) RUN NO(S)	201 202

The following messages are output because the names "N-C4," "N-C5," and "TOLUENE" are not model species on either the LPARM or the LM2ARM. ("N-C5" and "N-C5" are emitted ROG species which are represented by "C4C5" in the LPARM model and by "C4C5-ALK" in the LM2ARM model. "TOLUENE" is represented by "AAR1" in the LPARM model and by "TOLU" in the LM2ARM model.) No plots are output.

CALC 202: NAME = 'N-C4	' NOT FOUND
CALC 202: NAME = 'N-C5	' NOT FOUND
CALC 202: NAME = 'TOLUENE	' NOT FOUND
CALC 201: NAME = 'N-C4	' NOT FOUND
CALC 201: NAME = 'N-C5	' NOT FOUND
CALC 201: NAME = 'TOLUENE	' NOT FOUND
ENTER (2) RUN NO(S)	<cr>
CURVE 1:	C C4C5-ALK TOLU
CURVE 2:	C C4C5 AAR1
CURVE 3:	<cr>
ENTER (2) RUN NO(S)	201 202

TABLE 12. EXAMPLES OF PLOT MODE INPUT OUTPUT (CONCLUDED).

Prompt or output produced	User input
The following messages are output. No plots are produced.	
CALC 202: NAME = 'C4C5 ' NOT FOUND CALC 202: NAME = 'AAR1 ' NOT FOUND CALC 201: NAME = 'C4C5-ALK ' NOT FOUND CALC 201: NAME = 'TOLU ' NOT FOUND	
ENTER (2) RUN NO(S)	202,201
Two plots are produced. The first has the model species "C4C5-ALK" from calculation 202 plotted on the same plot as the species named "C4C5" from calculation 201, and the second has "TOLU" from calculation 202 plotted with "AAR1" from calculation 201. No warning messages are output.	
ENTER (2) RUN NO(S)	QUIT

SECTION V

EXAMPLE SCENARIOS AND REACTIVITY ASSESSMENT CALCULATIONS

In this section, several examples of the types of scenarios which can be represented in this model, and of reactivity assessment calculations which can be carried out using the User Interface are described. The scenarios and composition files used in these example calculations are all included in the initial distribution of this model, and the user can easily duplicate these example calculations himself, or he can use these scenarios and composition files as a starting point for carrying out calculations for other scenarios or mixtures. The example scenarios included with the initial distribution of this model, and the User Interface commands used to create them, are given in Section V.A, below, and example calculations with several representative test mixtures are described in Section V.B.

A. EXAMPLES OF SCENARIOS FOR REACTIVITY ASSESSMENT

A total of five idealized model scenarios, which can be used for the purpose of assessing reactivities of Air Force fuels and other test mixtures, are included with the initial distribution of this model. These are designated EKMA1, EKMA1, EKMA2, EKM1A, and EKM1B. All these scenarios are based on the standard city-specific EKMA inputs recommended by Gipson and Freas (Reference 6) of the U.S. EPA for use in Regulatory Impact Analyses for the national ozone standard to be applied in various regions of the United States. These are designed to represent, though not replace, input data for the various city-specific EKMA models used in State Implementation Plans. For this purpose, Gipson and Freas (Reference 6) divided the country into three regions on the basis of the mixing heights (i.e., the maximum amount of dilution occurring throughout the day), and specified inputs for EKMA model calculations designed to represent the conditions of those three regions. In the calculations reported here, the EKMA1 scenarios are based on the input data Gipson and Freas (Reference 6) recommend for the low dilution region (which they designate Region I), the EKMA3 scenarios employ the mixing height schedule they

recommend for Region III, the highest dilution region, and the EKMA2 scenario represents the middilution region. These are designated the "standard EKMA" scenarios in the subsequent discussion. The EKM1A and EKM1B scenarios are the same as the EKMA1 scenario except for the emissions schedule used for the test organics, which is discussed below.

All of these "EKMA" scenarios are 1-day simulations [starting at 0800 Local Daylight Time (LDT) and ending at 2100 LDT], with a fraction of the reactive organics and NO_x assumed to be present initially and with the remainder being emitted at a constant rate until 1500 LDT, when it is assumed that the air parcel is transported out of the source region. The inversion height is assumed to increase throughout the day based roughly on the "characteristic curve" used as the default in the OZIPP software (Reference 9) with the main difference between these two scenarios being the maximum inversion height. The scenarios also differ slightly in their assumed latitude, with the EKMA1, EKMA2, and EKMA3 scenarios using assumed latitudes of 34.1, 38.6, and 39.75, based on representative cities in each of the regions. The date for the scenarios is June 21, and the temperature is constant at 300 K. 56.5 percent of the initial NO_x and 67.1 percent of the initial base case ROG are assumed to be present initially.

For all these scenarios, the composition file used for the default base case ROG mixture in all scenarios is called SAISURG (SAIS on the Cyber), and the composition file used for the aloft mixture is designated EKMAALO (EKMAA on the Cyber), as discussed below. No background composition file is used; since relatively large fractions of the emitted ROG and NO_x are assumed to be present initially, background species would have little effect on these scenarios. The total ROG emissions level in all scenarios is arbitrarily held constant at 12.0 millimoles carbon m^{-2} day^{-1}, and the NO_x level is used as the varied parameter, with four levels, ranging from 3.0 to 0.3 millimoles m^{-2} day^{-1}. These correspond to base case ROG/NO_x ratios of 4, 8, 15, and 40, which represent NO_x conditions ranging from NO_x-rich to NO_x-poor, respectively. The NO_x emissions are assumed to consist of 25 percent NO_2 and 75 percent NO. Emissions are assumed to occur at a constant flux until 1430 solar daylight time, and then they decrease to 0 an hour later. In the "standard" EKMA scenarios, the test ROG mixture emissions is assumed to

have the same schedule as the base case emissions, with 67.1 percent of the test ROG species being present initially, as is the case with the base case ROG mixture. No default test mixture composition or level are included in these scenarios, since these are intended to be specified in the CALC command, as discussed in the following section.

As indicated above, the mixture given in the composition file SAISURG (VAX) or SAIS (Cyber) is used to represent the base case ROG emissions. This mixture is based on the organic surrogate mixture designed by the researchers at Systems Applications, Inc. (SAI) to represent current emissions into the California South Coast Air Basin. In terms of "Carbon Bond" species (Reference 10) the composition of this surrogate is almost identical with that recommended by the EPA for use in EKMA calculations in cases where no composition data are available (Reference 5) and this mixture has been employed in a number of environmental chamber studies as a representative of current urban ROG emissions (References 11-13). The composition of this mixture is given in Table 13. Note that the values are given in terms of ppm constituent per ppmC of mixture total, which is the appropriate units for normalized composition files for ROG mixtures.

TABLE 13. BASE CASE ROG AND ALOFT MIXTURES USED IN THE STANDARD EKMA SCENARIOS.

Species	Base case ROG mixture (ppb species/ppmC mix)	Aloft mixture (ppb)
Ozone		70.0
n-Butane	37.5	2.51
n-Pentane	40.0	
iso-Octane	18.75	
Ethene	25.0	0.60
Propene	16.7	0.30
Benzene		0.125
Toluene	17.86	0.375
m-Xylene	15.63	
Formaldehyde	37.5	1.50
Acetaldehyde		1.50
Unreactive	112.5	

As the inversion height increases, pollutants present in the air mass aloft are mixed into the ground layer and can affect the reactions occurring there. The composition of pollutants aloft assumed in the standard EKMA scenarios is specified in the EKMAALO composition file (EKMAA on Cybers), and is based on recommendations of Gipson,* Gipson and Freas (Reference 6), and the EPA guidelines (Reference 5). Gipson and Freas (Reference 6) recommend assuming 0.07 ppm ozone aloft, and Gipson* considers approximately 20 ppbC of ROG and negligible NO_x aloft to be typical of many situations, and this was assumed in both these scenarios. The composition of the aloft ROG is based on the recommendation given by the EPA (Reference 5) for use as defaults when employing the Carbon Bond mechanism (Reference 10) for EKMA calculations; since these are given in terms of carbon bond species, they were translated into the representative molecular species utilized by this model. The resulting composition used for aloft mixtures in the standard EKMA scenarios is given in Table 13.

The output options included with these scenarios as included with the model are to produce a full output listing file, and to produce both plots and tabulations of ozone, NO, NO_2, PAN, and formaldehyde with output files not spooled to the printer. Because the full output listing file is rather large, if the user plans to employ these scenarios, or scenarios based on these as starting points, for a large number of calculations, it is recommended that he specify no Integration program output (OUT=NO), and include in the plots and tabulations any other model compounds whose calculated levels are of interest, if any.

To illustrate how this model can be used to assess effects of the location of emissions sources of the test mixtures on their calculated reactivities, three versions of each of these EKMA scenarios, which differ only in their emissions schedules for test organics (TMS), are included with this model. In the standard EKMA scenarios, the emissions schedule for the test mixture is exactly the same as that for the base organics (i.e., TMS = BOS), to illustrate the case when the test mixture is emitted at the same time with the rest of the organics. The EKM1A and EKM1B

---

*Gipson, G. L., U. S. Environmental Protection Agency, personal communication, 1985.

scenarios are used to represent the effects of differing assumptions concerning the schedule for test mixture emissions. In the EKM1A scenario, all the test mixture is assumed to be present initially, to represent a case when the air parcel passes over the source of the test mixture prior to entering the airshed being simulated. In the EMA1B scenario, the test mixture emissions schedule is designed to represent cases where the air parcel passes over the source of the test mixture after going through most of the base case emissions source area. In this scenario, all emissions of test organics are represented as occurring in a 1-hour period, between 1400 and 1500.

A series of MAIN, SCEN, and COMP mode commands which can be used to create all the scenarios and test mixtures described above is shown in Table 14. (The prompts and the user input are given in separate columns for easier readability; when the program is actually run, they are generally separated by one or no spaces.) Note that the EKMA1 scenario is created from scratch using the ENTER option of the SCEN mode, while the other scenarios are created by specifying the EKMA1 scenario as the starting point, and then entering only those scenario options which are different. The table includes comments indicating what is being entered. The scenarios are entered in the order EKMA1, EKMA2, EKMA3, EKM1A, and EKM1B, and the composition files are entered in the order SAISURG and EKMAALO, and the longer (VAX) file names are used in this example.

The output produced by the program when the LIST ALL command is given when the EKMA1 scenario is in memory is shown in Table 15. The listings for the other scenarios are similar.

B. CALCULATED REACTIVITIES OF REPRESENTATIVE TEST MIXTURES

This section illustrates how this model, and the example scenarios discussed in the previous section, can be used for carrying out reactivity assessment calculations. In addition to the scenarios and associated base case ROG and aloft species composition files, discussed above, the initial distribution of this model also includes composition files of seven test mixtures, six representing the compositions of the various "synthetic"

TABLE 14. USER INTERFACE COMMANDS USED TO CREATE SCENARIOS AND COMPOSITION FILES FOR REACTIVITY ASSESSMENT EXAMPLES.

Prompts and program output	User input
---------- Define Scenario EKMA1 ----------[a]	
MAIN:	SCEN EKMA1
Scenario "EKMA1 " previously undefined Scenario to read (? to see old filenames, <cr> if none):	<cr>
SCEN: Enter X if not-applicable, Q to return to SCEN: or HELP	ENTER
-------- General Scenario Parameters --------	
Model Name: ( )	LPARM
Varied Parameter: ( )	NOX
Pollutant Level Units M)olar or P)pm: (U)	M
---------- Physical Parameters ----------	
Daylight Time? N)o or Y)es: (U)	Y
Latitude (e.g. 34.10): (-1.000)	34.10
Date mm.dd (e.g. 10.31): (-1.000)	6.21
Start Time: (-1)	800
End Time: (-1)	2100
Save Interval: (-1.000)	60
Light Factor: (1.)	<cr>
Inversion Height: (UNDEF)	SCHED
Temperature: (300.)	<cr>
Relative Humidity: (50.)	<cr>
-------- Chemical Species Parameters --------	
Background Mix ID: ( )	<cr>
Background Mix Level:(1.)	<cr>
Aloft Mix ID: ( )	EKMAALO
NOX Level: (VARY)	<cr>
NXS (NOX Schedule). Enter SCHED or NONE: (NONE)	SCHED
NO2 Fraction: (-1.000)	0.25
Base Organics Comp. ID: ( )	SAISURG
Base Organics Level: ( )	12.0
Base Organics Schedule. Enter SCHED or NONE: (NONE)	SCHED
Test Mixture Level: ( )	<cr>
Test Mixture ID: ( )	<cr>
Test Mixture Schedule. Enter SCHED or NONE: (NONE)	SCHED

TABLE 14. USER INTERFACE COMMANDS USED TO CREATE SCENARIOS AND COMPOSITION FILES FOR REACTIVITY ASSESSMENT EXAMPLES (CONTINUED).

Prompts and program output	User input
--------- Inversion Height Schedule ----------	
SCHED IH :	TY=L
SCHED IH :	552,250
SCHED IH :	651,250
SCHED IH :	1247,492
SCHED IH :	1613,600
SCHED IH :	2100,600
SCHED IH :	<cr>
---------- NO_x Emissions Schedule ----------	
SCHED NXS:	TY=L
SCHED NXS:	IF=.565
SCHED NXS:	800,1
SCHED NXS:	1430,1
SCHED NXS:	1530,0
SCHED NXS:	2100,0
SCHED NXS:	<cr>
-------- Base ROG Emissions Schedule ---------	
SCHED BOS:	TY=L
SCHED BOS:	IF=.671
SCHED BOS:	800,1
SCHED BOS:	1430,1
SCHED BOS:	1530,0
SCHED BOS:	2100,0
SCHED BOS:	<cr>
--------- Test ROG Emissions Schedule ----------	
SCHED TMS:	TY=L
SCHED TMS:	IF=.671
SCHED TMS:	800,1
SCHED TMS:	1430,1
SCHED TMS:	1530,0
SCHED TMS:	2100,0
SCHED TMS:	<cr>
--------- Variable NO_x Level Values ----------	
Varied parameter. Enter VPn,value:	VP,3.0
Varied parameter. Enter VPn,value:	VP,1.5
Varied parameter. Enter VPn,value:	VP,.75
Varied parameter. Enter VPn,value:	VP,.3
Varied parameter. Enter VPn,value:	<cr>

TABLE 14. USER INTERFACE COMMANDS USED TO CREATE SCENARIOS AND COMPOSITION FILES FOR REACTIVITY ASSESSMENT EXAMPLES (CONTINUED).

Prompts and program output	User input
------ Non-Standard Parameters Not Used -------	
Non-standard parameter NSPn,name,number:	<cr>
---------- Output Option Values ----------	
Integration Output List File (YN): (Y)	Y
Plots of Calc'd Results (YN): (N)	Y
Tables of Calc'd Results (YN): (N)	Y
Derivatives on Output List File (YN): (N)	N
Rates on Output List File (YN): (N)	N
Full Model Listing(Reaction,Species)(YN): (N)	N
Spool Output File to Printer (YN): (N)	N
-------- Species to Plot and Tabulate --------	
Plot/print species. Enter PSn,name:	PS,O3
Plot/print species. Enter PSn,name:	PS,NO
Plot/print species. Enter PSn,name:	PS,NO2
Plot/print species. Enter PSn,name:	PS,C-PAN
Plot/print species. Enter PSn,name:	PS,HCHO
Plot/print species. Enter PSn,name:	<cr>
Enter INCOM to see if any variables are undefined	
SCEN:	SAVE
SCENARIO COMPLETE Scenario EKMA1 saved	
------------ Define Scenario EKMA2 ----------	
MAIN:	EKMA2
Scenario "EKMA2" previously undefined Scenario to read (? to see old filenames, <cr> if none):	EKMA1
Scenario EKMA1 read Varied parm = NOX Enter LIST ALL or HELP for more information	
---------- Change Latitude ----------	
SCEN:	LAT=38.6

TABLE 14. USER INTERFACE COMMANDS USED TO CREATE SCENARIOS AND COMPOSITION FILES FOR REACTIVITY ASSESSMENT EXAMPLES (CONTINUED).

Prompts and program output	User input
------ Change Inversion Height Schedule -------	
SCEN:	SCHED IH
SCHED IH :	NEW
SCHED IH :	552,250
SCHED IH :	651,250
SCHED IH :	1247,1100
SCHED IH :	2100,3500
SCHED IH :	<cr>
SCEN:	SAVE
SCENARIO COMPLETE	
Scenario EKMA2 saved	
------------ Define Scenario EKMA3 ----------	
MAIN:	SCEN EKMA3
Scenario "EKMA3" previously undefined	
Scenario to read (? to see old filenames, <cr> if none):	EKMA1
Scenario EKMA1 read	
Varied parm = NOX	
Enter LIST ALL or HELP for more information	
------------- Change Latitude ----------	
SCEN:	LAT 39.75
------ Change Inversion Height Schedule -------	
SCEN:	SCHED IH
SCHED IH :	NEW
SCHED IH :	552,250
SCHED IH :	651,250
SCHED IH :	1247,2490
SCHED IH :	1613,3500
SCHED IH :	2100 3500
SCHED IH :	<cr>
SCEN:	SAVE
SCENARIO COMPLETE	
Scenario EKMA3 saved	

TABLE 14. USER INTERFACE COMMANDS USED TO CREATE SCENARIOS AND COMPOSITION FILES FOR REACTIVITY ASSESSMENT EXAMPLES (CONTINUED).

Prompts and program output	User input
---------- Define Scenario EKM1A ----------	
MAIN:	SCEN EKM1A
Scenario "EKM1A" previously undefined Scenario to read (? to see old filenames, <cr> if none):	EKMA1
Scenario EKMA1 read Varied parm = NOX Enter LIST ALL or HELP for more information	
----- Change Test Mixture Initial Fraction ----	
SCEN: SCHED TMS: SCHED TMS: SCEN:	SCHED TMS IF=1. <cr> SAVE
SCENARIO COMPLETE Scenario EKM1A saved	
---------- Define Scenario EKM1B ----------	
MAIN:	SCEN EKM1B
Scenario "EKM1B" previously undefined Scenario to read (? to see old filenames, <cr> if none):	EKMA1
Scenario EKMA1 read Varied parm = NOX Enter LIST ALL or HELP for more information	
--------- Change Test Mixture Schedule --------	
SCEN: SCHED TMS: SCHED TMS: SCHED TMS: SCHED TMS: SCHED TMS: SCHED TMS:	SCHED TMS IF=0 NEW 1400,0 1430,1 1500,0 <cr>
SCEN: SAVE	
SCENARIO COMPLETE Scenario EKM1B saved	

TABLE 14. USER INTERFACE COMMANDS USED TO CREATE SCENARIOS AND COMPOSITION FILES FOR REACTIVITY ASSESSMENT EXAMPLES (CONCLUDED).

Prompts and program output	User input
---------- Define Composition SAISURG ----------	
MAIN:	COMP SAISURG
SAISURG is a new composition file Composition filename to read (? to see old filenames. <cr> if none)	<cr>
------- Input Compositions as ppb/ppmC --------	
COMP:	N-C4 37.5
COMP:	N-C5 40.0
COMP:	ISO-C8 18.75
COMP:	ETHENE 25.0
COMP:	PROPENE 16.7
COMP:	TOLUENE 17.86
COMP:	M-XYLENE 15.63
COMP:	FORMALD 37.5
COMP:	INERT 112.5
----- Normalize to yield 1 Carbon Total ------	
COMP:	NORM
COMP:	SAVE
---------- Define Composition EKMAALO ----------	
MAIN:	COMP EKMAALO
EKMAALO is a new composition file Composition filename to read (? to see old filenames. <cr> if none)	<cr>
------- Input Compositions as ppm --------	
COMP:	O3 0.070
COMP:	N-C4 0.00251
COMP:	ETHENE 0.00060
COMP:	PROPENE 0.00030
COMP:	BENZENE 0.000125
COMP:	TOLUENE 0.000375
COMP:	FORMALD 0.00150
COMP:	ACETALD 0.00150
COMP:	SAVE

[a]Comments highlighted by dashes ("----comment----") are given to explain what is being done, and are not part of the program input or output.

TABLE 15. LISTING OF "EKMA1" EXAMPLE SCENARIO, AS PRODUCED BY THE "LIST ALL" SCEN MODE COMMAND.

Scenario = EKMA1 Type = AIRSHED. 2 Parameter(s).
Scenario status = COMPLETE . General parms:

CODE	VALUE	DESCRIPTION
(MOD)	LPARM	Model Name
(VAR)	NOX	Varied parameter
(PLU)	MOLAR	Pollutant Level Input Units (MOLAR = millimoles/m2) (PPM = ppm at initial IH)
(DT)	YES	Daylight Time? (Yes/No)
(LAT)	34.10	Latitude (positive=North)
(DAT)	6.21	Date (mm.dd)
(ST)	800	Start Time (HHMM)
(ET)	2100	End Time (DHHMM)
(SA)	60.00	Save Interval (min)
(LF)	1.	Light Factor
(IH)	SCHED	Inversion Height (meters)
(TEM)	300	Temperature (K)
(RH)	50	Humidity (%)

Chemical Species

CODE	VALUE	DESCRIPTION
(BMC)		Background Mix ID
(BML)	1.	Background Mix Level
(ALO)	EKMAALO	Aloft Mix ID
(NOX)	VARY	NO_x Level
(NXS)	SCHED	NO_x Schedule
(NO2)	0.2500	NO_2 Fraction
(BOC)	SAISURG	Base Organics Comp ID
(BOL)	12.	Base Organics Level
(BOS)	SCHED	Base Organics Schedule
(TMC)		Test Mixture ID
(TML)		Test Mixture Level
(TMS)	SCHED	Test Mixture Schedule

LF not referenced as SCHED

SCHEDULE = IH , No. times = 5
Schedule Status = COMPLETE

TABLE 15. LISTING OF "EKMA1" EXAMPLE SCENARIO, AS PRODUCED BY THE "LIST ALL" SCEN MODE COMMAND (CONTINUED).

```
CODE    VALUE     DESCRIPTION
----    -----     -----------------
(TY)    LINEAR    Schedule Type: L)inear/S)tepped

 TIME    VALUE     TIME    VALUE     TIME    VALUE     TIME    VALUE
 ----    -----     ----    -----     ----    -----     ----    -----
  552 2.50E+02      651 2.50E+02     1247 4.92E+02     1613 6.00E+02
 2100 6.00E+02

Times are given as clock times (DHHMM).


TEM not referenced as SCHED


SCHEDULE =     NXS,  No. times =  4
                              Schedule Status = COMPLETE

CODE    VALUE     DESCRIPTION
----    -----     -----------------
(TY)    LINEAR    Schedule Type: L)inear/S)tepped
(IF)    .565      Initial Fraction (NORMALIZED SCHEDULES ONLY)

 TIME    VALUE     TIME    VALUE     TIME    VALUE     TIME    VALUE
 ----    -----     ----    -----     ----    -----     ----    -----
  800    1.0       1430    1.0       1530 0.00E+00     2100 0.00E+00

Times are given as clock times (DHHMM).


SCHEDULE =     BOS,  No. times =  4
                              Schedule Status = COMPLETE

CODE    VALUE     DESCRIPTION
----    -----     -----------------
(TY)    LINEAR    Schedule Type: L)inear/S)tepped
(IF)    .671      Initial Fraction (NORMALIZED SCHEDULES ONLY)

 TIME    VALUE     TIME    VALUE     TIME    VALUE     TIME    VALUE
 ----    -----     ----    -----     ----    -----     ----    -----
  800    1.0       1430    1.0       1530 0.00E+00     2100 0.00E+00

Times are given as clock times (DHHMM).


SCHEDULE =     TMS,  No. times =  4
                              Schedule Status = COMPLETE
```

TABLE 15. LISTING OF "EKMA1" EXAMPLE SCENARIO, AS PRODUCED BY THE "LIST ALL" SCEN MODE COMMAND (CONCLUDED).

CODE	VALUE	DESCRIPTION
(TY)	LINEAR	Schedule Type: L)inear/S)tepped
(IF)	.671	Initial Fraction (NORMALIZED SCHEDULES ONLY)

TIME	VALUE	TIME	VALUE	TIME	VALUE	TIME	VALUE
800	1.0	1430	1.0	1530	0.00E+00	2100	0.00E+00

Times are given as clock times (DHHMM).

Varied parameter values. Varied parameter = NOX

Calc	Value	Calc	Value	Calc	Value	Calc	Value
(VP1)	3.	(VP2)	1.5	(VP3)	.75	(VP4)	.3

Nonstandard Scenario Parameters. Number = 0

CODE	PARAMETER NAME	VALUE (if applicable)
(NSP1)	NO DATA	

Output options

CODE	VALUE	DESCRIPTION
(OUT)	YES	Produce integration output listing file
(PLT)	YES	Produce plots of calculated results
(TAB)	YES	Produce tabulations of calculated results
(WDY)	NO	Summarize derivatives on output listing
(WRA)	NO	Summarize rates on output listing
(WRM)	NO	Produce full model listing(rates,species)
(SPO)	NO	Spool output files to system printer

Output Species to be Plotted and/or Tabulated

CODE	NAME	CODE	NAME	CODE	NAME
(PS1)	O3	(PS2)	NO	(PS3)	NO2
(PS4)	C-PAN	(PS5)	HCHO		

turbine engine fuels used in the experiments carried out to test the chemical mechanism in the model, and one representing a synthetic jet exhaust mixture also used in the experiments to test this model (References 1, 7). In this example, we illustrate calculations of effects of addition of these test mixtures to the EKMA1 and the EKMA3 scenarios. The results will be given in terms of the effects on the calculated daily maximum ozone level, but the effects on other measures of air quality, such as, for example, maximum PAN or formaldehyde levels, or on lifetimes of emitted pollutants, etc., could also be derived from the results of these example calculations.

The composition files associated with the surrogate fuel test mixtures are designated FUEL1, FUEL2, FUEL3, FUEL1F, FUEL1T, and FUEL1P (the latter three being FUL1F, FUL1T, FUL1P on Cybers). All six mixtures represent synthetic model fuels employed for which environmental chamber experiments were carried out as part of the experimental program used to develop and test this model, and are described in a previous report to the USAF (Reference 7). Briefly, the composition of FUEL1 was designed by USAF scientists to represent JP-4 fuels currently in use; FUEL2 is similar to FUEL1 but the total aromatic content is increased, and FUEL3 is also similar to FUEL1 except that the ratio of alkylbenzenes to bicyclic aromatics was increased, while keeping the total aromatic content the same. FUEL1F, FUEL1T, and FUEL1P are the same as FUEL1, except that they contain 1 percent (on a mole carbon basis) of furan, thiophene, or pyrrole, respectively, to represent fuels with these potential impurities. The compositions of these different synthetic fuel mixtures are summarized in Table 16.

An additional composition file, SYNEXH, is included in these examples of test mixtures. This mixture was designed by scientists at AFESC/RDVS to represent the composition of jet exhausts (Reference 14), and was used in chamber experiments carried out to test this model (Reference 1). The composition of this mixture is also shown in Table 16.

In the example calculations, we will examine the effects on the maximum ozone concentrations in the EKMA1 and EKMA3 scenarios caused by adding a given amount of the test mixtures, arbitrarily set at 10 percent of the total base-case ROG mixture. This requires "base-case"

TABLE 16. COMPOSITIONS OF THE REPRESENTATIVE TEST MIXTURES USED IN THE REACTIVITY ASSESSMENT EXAMPLES.

	ppb Compound/ppmC mixture						
Compound	FUEL1	FUEL2	FUEL3	FUEL1F	FUEL1T	FUEL1P	SYNEXH
N-C6	18.98	16.50	19.48	18.79	18.79	18.79	
N-C7	25.40	21.71	25.97	25.15	25.15	25.15	
N-C8	23.36	19.88	23.85	23.12	23.12	23.12	9.80
N-C10	2.57	2.07	2.36	2.54	2.54	2.54	
N-C14							1.70
CYC-C6	10.82	9.33	10.99	10.72	10.72	10.72	
ME-CYCC6	17.84	15.29	18.27	17.66	17.66	17.66	7.50
ET-CYCC6	3.12	2.50	3.00	3.09	3.09	3.09	
BENZENE							4.80
TOLUENE	11.27	16.86	14.56	11.16	11.16	11.16	4.10
STYRENE							0.85
M-XYLENE							5.80
P-XYLENE	2.50	3.75	3.25	2.47	2.47	2.47	
1-C3-BEN	2.11	2.89	2.55	2.09	2.09	2.09	
135-TMB	3.33	4.56	3.89	3.30	3.30	3.30	
TETRALIN	4.40	5.90	2.70	4.35	4.35	4.35	
NAPHTHAL	3.00	4.20	1.80	2.97	2.97	2.97	1.20
ME-NAPH	3.27	4.09	1.91	3.24	3.24	3.24	0.85
DM-NAPH	1.08	1.42	0.67	1.07	1.07	1.07	
ETHENE							137.00
PROPENE							37.00
1-BUTENE							16.60
1-HEXENE							16.38
FORMALD							36.00
ACETALD							13.60
ACROLEIN							5.80
MEGLYOX							2.90
FURAN				2.50			
THIOPHEN					2.50		
PYRROLE						2.50	
INERT							75.44

calculations, i.e., calculations with no added test mixture, for both scenarios, and calculations for each scenario and each test mixture where the test mixture is emitted at a level of 1.2 millimole C m^{-2} day^{-1}, which is 10 percent of the base case ROG emissions level of 12 millimole C m^{-2} day^{-1}.

Assuming that the necessary scenarios and composition files have been created and saved, then these calculations can be initiated by giving the following commands in the MAIN mode of the User Interface. Note that the comments are included so that the calculation label will indicate that the calculations were carried out using the LPARM model. (The label field of the CALC command does not need to indicate the scenario, test mixture, or test mixture level, because the program automatically adds this information to the calculation label if they were specified on the CALC command line.)

```
CALC EKMA1 "BASE CASE.  LPARM MODEL EXAMPLE"
CALC EKMA1 FUEL1 1.2 "LPARM MODEL EXAMPLE"
CALC EKMA1 FUEL2 1.2 "LPARM MODEL EXAMPLE"
CALC EKMA1 FUEL3 1.2 "LPARM MODEL EXAMPLE"
CALC EKMA1 FUEL1F 1.2 "LPARM MODEL EXAMPLE"
CALC EKMA1 FUEL1T 1.2 "LPARM MODEL EXAMPLE"
CALC EKMA1 FUEL1P 1.2 "LPARM MODEL EXAMPLE"
CALC EKMA1 SYNEXH 1.2 "LPARM MODEL EXAMPLE"
CALC EKMA3 "BASE CASE.  LPARM MODEL EXAMPLE"
CALC EKMA3 FUEL1 1.2 "LPARM MODEL EXAMPLE"
CALC EKMA3 FUEL2 1.2 "LPARM MODEL EXAMPLE"
CALC EKMA3 FUEL3 1.2 "LPARM MODEL EXAMPLE"
CALC EKMA3 FUEL1F 1.2 "LPARM MODEL EXAMPLE"
CALC EKMA3 FUEL1T 1.2 "LPARM MODEL EXAMPLE"
CALC EKMA3 FUEL1P 1.2 "LPARM MODEL EXAMPLE"
CALC EKMA3 SYNEXH 1.2 "LPARM MODEL EXAMPLE"
EXIT
```

The program responds after each CALC command with a message giving the calculation number of the last calculation and the number of calculations

created by the command, which in this is four, since there are four values of the varied parameter NOX in both these scenarios. The last command terminates the User Interface program and causes the calculations to be submitted to the batch processor, or run at the user's terminal, depending on the way the User Interface program was invoked (for Cyber systems), or the response given to the prompt after the EXIT command is input (for VAX systems), as discussed in Section IV.B.

After the calculations are completed, the user can examine the plots or tabulations produced in the Plot outputs for the calculation and determine the relative effects of the various test mixtures on the pollution levels for the various scenarios. For example, Table 17 shows the maximum ozone concentrations calculated for the EKMA1 and EKMA3 base case simulations, and the increase in maximum ozone concentration caused by adding each of the test mixtures to the emissions. Since the amounts added in each case are the same, these results indicate the relative reactivities of each of the test mixtures with respect to maximum ozone formation, and how these vary with the scenario and the ROG/NO_x ratio.

TABLE 17. MAXIMUM BASE CASE OZONE LEVELS CALCULATED FOR THE EXAMPLE EKMA SCENARIO, AND CALCULATED CHANGES IN MAXIMUM OZONE CAUSED BY THE ADDITION OF REPRESENTATIVE TEST MIXTURES.

Scen.	ROG/NO_x	Base O_3 (ppb)	Change in maximum ozone (ppb)[a]						
			FUEL1	FUEL2	FUEL3	FUEL1F	FUEL1T	FUEL1P	SYNEXH
EKMA1	4	67	7.6	9.0	7.9	7.9	7.6	7.8	32.2
	8	204	9.1	9.6	9.4	9.3	9.0	9.2	26.4
	16	186	1.0	0.7	1.1	1.1	1.0	1.0	7.2
	40	132	-1.1	-1.3	-1.0	-1.1	-1.1	-1.1	1.6
EKMA3	4	82	3.4	3.8	3.5	3.5	3.4	3.5	11.7
	8	98	1.6	1.7	1.7	1.7	1.6	1.7	4.6
	16	88	0.6	0.6	0.6	0.6	0.6	0.6	2.5
	40	74	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1	0.8

[a]Added test mixture = 10 percent of base case ROG emissions.

The results of these calculations show that the effect of the addition of these mixtures depends significantly on the ROG/NO_x ratio, being the greatest (on a percentage basis) at the lowest ratios. Indeed, the fuels are negatively reactive at the highest ROG/NO_x ratios; this can be attributed to the NO_x sinks in the reaction mechanisms of aromatic hydrocarbons and high molecular weight alkanes (Reference 2), which are the major constituents of these fuels. These results also show that the synthetic exhaust mixture is much more reactive, at least in terms of atmospheric ozone formation, than are the synthetic fuels. This is consistent with the results of the chamber experiments carried out with these mixtures, where much lower levels of the synthetic exhaust mixture was required to achieve ozone formation during a 6-hour irradiation than was the case of any of the whole or synthetic fuels studied (Reference 1). On the other hand, it is interesting to note that there appears to be relatively little difference between the synthetic fuels in their reactivities in these scenarios, even among those which are quite different in reactivity when irradiated by themselves in the environmental chamber experiments (Reference 7). For example, the addition of ~1 percent furan or pyrrole to the "standard" fuel had a very large effect on the rate of ozone formation when the fuel is irradiated by itself in the chamber experiments (Reference 7), yet it has a relatively small effect on maximum ozone yields under the conditions of these scenarios, even at the lowest ROG/NO_x ratios. This illustrates that relative reactivities of test mixtures under atmospheric conditions might be quite different than observed when irradiated in chamber experiments. The utility of this model is that it can be used to estimate relative reactivities under conditions which are potentially much more representative of conditions of real airsheds than would be practical in any type of chamber experiment.

The user can obtain an indication of the dependence of the relative reactivities of these test mixtures on other scenario conditions beside those varied in these examples by carrying out exactly analogous calculations for the other scenarios which are included with this model. For example, calculations using the EKM1A and EKM1B scenarios could be employed to determine how the emissions schedule for the test mixture

affects its reactivity. (Note that in this case, it would not be necessary to repeat the base case simulations, since the base case for EKM1A and EKM1B are exactly the same as for EKMA1, since they differ only in the test mixture emissions schedule.) The SCEN command could be used to create new scenarios to determine dependences of reactivity on other aspects of the scenario, such as, for example, the composition of the base case ROG emissions. The user can also determine the relative reactivities of other test mixtures for these scenarios by creating new composition files using the COMP feature, and then using them as the test mixture in the CALC commands with these scenarios.

This example focuses on the applications of this modeling system for reactivity assessment purposes, since the level of detail of the chemical model it employs makes it particularly suitable for this purpose. However, this model can obviously be employed for other applications where box model airshed simulations might be useful, such as carrying out EKMA analyses (References 4, 5), assessing effects of proposed control strategies, assessing atmospheric impacts of proposed new emissions sources, or assessing the contribution of one particular emissions source (such as, for example, an Air Force base) on overall pollution levels in surrounding areas. Obviously in these cases the scenario should be designed to be as representative of the specific airshed being simulated as possible, and the idealized scenarios used in these examples might not be appropriate. A discussion of the most appropriate scenarios for all the possible applications of this model is beyond the scope of this document.

REFERENCES

1. Carter, W.P.L., Winer, A.M., Atkinson, R., Heffron, S.E., Poe, M.P., and Goodman, M.A., Atmospheric Photochemical Modeling of Turbine Engine Fuels, Phase II, Computer Model Development, Draft Final Report, Engineering & Services Laboratory, Air Force Engineering & Services Center, Tyndall AFB, FL, April 1987.

2. Carter, W.P.L. and Atkinson, R., A Computer Modeling Study of Incremental Hydrocarbon Reactivity, Final Report to EPA Cooperative Agreement No. CR810214, Atmospheric Sciences Research Laboratory, Office of Research and Development, U. S. EPA, Research Triangle Park, NC, March 1987.

3. Zannetti, P., "A Short Course on Air Pollution Modeling," Computational Mechanics Centre, Southampton, England, 1-2 December, 1980.

4. Dodge, M.C. "Combined Use of Modeling Techniques and Smog Chamber Data to Derive Ozone-Precursor Relationships," in Proceedings, International Conference on Photochemical Oxidant Pollution and Its Control, EPA-300/3-77-001, 1977.

5. Gipson, G.L., Freas, W.P., Kelly, K.R., and Meyer, E.L., Guideline for Use of City-Specific EKMA in Preparing Ozone SIP's, EPA-450/4-80-027, March 1981.

6. Gipson, G.L. and Freas, W.P., Use of City-Specific EKMA in the Ozone RIA, U. S. Environmental Protection Agency, July 1983.

7. Carter, W.P.L., Winer, A.M., Atkinson, R., Dodd, M.C., Long, W.D., and Aschmann, S.M., Atmospheric Photochemical Modeling of Turbine Engine Fuels, Phase I. Experimental Studies, Vol. I, Results and Discussion, ESL-TR-84-32, Engineering & Services Laboratory, Air Force Engineering & Services Center, Tyndall Air Force Base, FL, July 1984.

8. Lurmann, F.W., Carter, W.P.L., and Coyner, L.A., A Surrogate Species Chemical Reaction Mechanism for Urban-Scale Air Quality Simulation Models. Volume 1 - Adaptation of the Mechanism, Final Report to EPA Contract No. 68-02-4104, Atmospheric Sciences Research Laboratory, Office of Research and Development, U.S. EPA Research Triangle Park, NC, 1987.

9. Gipson, G.L., Users Manual for OZIPM-2: Ozone Isopleth Plotting with Optional Mechanisms/Version 2, EPA-450/4-84-024, August 1984.

10. U.S. Environmental Protection Agency, Guideline for Using the Carbon Bond Mechanism in City Specific EKMA, EPA-450/4-84-005, February 1984.

11. Carter, W.P.L., Dodd, M.C., Long, W.D., and Atkinson, R., Outdoor Chamber Study to Test Multi-Day Effects, EPA-600/3-84-115.

12. Carter, W.P.L., Atkinson, R., Long, W.D., Parker, L.N., and Dodd, M.C., Effects of Methanol Fuel Substitution on Multi-Day Air Pollution Episodes, Final Report, California Air Resources Board Contract No. A3-125-32, September 1986.

13. Pefley, R.K., Pullman, J.B., and Whitten, G.Z., The Impact of Alcohol Fuels on Urban Air Pollution: Methanol Photochemistry Study, Final Report, U. S. Department of Energy, Office of Vehicle and Engine R&D, Washington, D.C., 1984.

14. Spicer, C.W., Holdren, M.W., Lyon, T.F., and Riggin, R.M., Composition and Photochemical Reactivity of Turbine Engine Exhaust, Final Report, F-98635-82-C-0131, Engineering and Services Laboratory, Air Force Engineering and Services Center, Tyndall Air Force Base, Florida, May 1984.

APPENDIX A

QUICK REFERENCE GUIDE FOR THE USER INTERFACE AND PLOTTING PROGRAMS

This Appendix gives summary tables of User Interface commands, Plotting program options, and ROG and model species names, which can be used to serve as a quick-reference guide for experienced users of this model. Most of these are shortened versions of more detailed tables given in Section III or IV of this manual. Table A-1 gives a summary of the MAIN mode User Interface commands; Table A-2 summarizes the scenario parameters; Table A-3 lists the SCEN mode commands; and Table A-4 summarizes the Comp mode commands. The PLOT mode and Plotting program options documented in this manual are summarized in Table A-5. Finally, Table A-6 lists the names of ROG species which can be included in composition files for ROG emissions in the current version of the model; and Table A-7 lists the nonemitted model compounds which can be included in composition files specifying background or aloft pollutants. The information given in Tables A-1 through A-5 is determined by the software, while the information in Tables A-6 and A-7 is determined by the chemical model.

In all tables summarizing program commands, ALL CAPS indicate commands or options which must be entered as shown, lower case indicates symbols which must be filled in by the user (such as names, values, etc.), and brackets ("[]") indicate optional input. If spaces are shown, they are part of the syntax (and only one space, not more, should be entered), though in most cases spaces can be replaced by a comma or an equal sign. Spaces should not be inserted where none are shown. Other punctuation must be entered as shown.

TABLE A-1. SUMMARY OF MAIN MODE COMMANDS FOR THE USER INTERFACE.

Command	Explanation
CALC ["label"]	Calculate the scenario in memory. Append the string "label" to the calculation label.
CALC scenario [tmc,tml]]["label"]	Calculate the named scenario, using "tmc" as the test mixture, and "tml" as the test mixture amount.
CALCNO	Display calculation numbers for last calculations.
COMP compfilename	Enter the COMP mode.
CRT	Pause during lengthy output (default).
DISPLAY OUTP calcno	Display Integration program listing output file.
DISPLAY PLOT calcno	Display plot program output file.
EXIT	Normal exit of program. Calculations submitted.
EXPL	Outputs an explanatory text file.
HELP [category]	Gives help display (HELP HELP displays categories).
INCOM	Lists undefined parameters for scenario in memory.
LIST option	List parameters for scenario in memory. Options are ALL, SCEN, PHYS, CHEM, VP, NSP, OUTP, SCHEDS.
LIST ALLCOMP	List all available composition filenames.
LIST ALLSCE	List all available scenario names.
LOG [calcno]	Give log of completed calculations starting at calculation number given.
PLOT	Enter the PLOT mode.
PRINTER	No pause during lengthy output.
QUIT	Quit program without submitting calculations.
SCEN	Enter the SCEN mode to edit scenario in memory.
SCEN scenario	Enter the SCEN mode to edit named scenario.

TABLE A-2. SUMMARY OF SCENARIO PARAMETERS.

Code	Value type[a]	Description
	General Scenario Parameters (SCEN)[b]	
MOD	model	Chemical MODel name.
VAR	parm	Name of the VARied parameter, if any. Optional.
PLU	code	Pollutant Level input Units (MOLAR or PPM).
	Physical Parameters (PHYS)[b]	
DT	y/n	Daylight Time used?
LAT	number	LATitude in degrees.
DAT	number	DATe, given as "month.day", e.g. 6.07 = June 7.
ST	clock	Start Time for the simulation.
ET	clock	End Time for simulation.
SA	number	Time interval to SAve data (minutes).
LF	nbr/sch	Light Factor. Can be scheduled. Can be varied.
IH	nbr/sch	Inversion Height in meters. Can be scheduled. Can be varied.
TEM	nbr/sch	TEMperature, in degrees K. Default=300. Can be scheduled.
RH	number	Relative Humidity in percent. Default=50. Can be varied.
	Chemical Species Parameters (CHEM)[b]	
BMC	p.cmp	Background Mix Composition filename. Optional.
BML	number	Background Mix Level factor. Default=1.
ALO	p.cmp	ALOft mix composition filename. Optional.
NOX	number	Total amount of NO_x emitted. Units depend on PLU. Can be varied.

TABLE A-2. SUMMARY OF SCENARIO PARAMETERS (CONTINUED).

Code	Value type[a]	Description
NXS	code	NO_x emissions Scheduled? SCHED=yes, NONE=no.
NO2	number	NO_2/NOx ratio.
BOC	e.cmp	Base Organics emissions Composition filename. Optional.
BOL	number	Amount (Level) of Base Organics emitted. Units depend on PLU.
BOS	code	Base Organics emissions Scheduled? SCHED=yes, NONE=no.
TMC	sp/ecmp	Test Mixture Composition or ROG species. Optional. Can be superseded by CALC command.
TML	number	Amount (Level) of Test Mixture emitted. Units depend on PLU. Can be superseded by CALC command.
TMS	code	Test Mixture emissions Scheduled? SCHED=yes, NONE=no.
		Varied Parameter Values (VP)[b]. Only used if VAR defined.
VP	value	Add new varied parameter value.
VPn	value	Replace the nth varied parameter value.
VPn,X		Delete the nth varied parameter value.
		Nonstandard Parameters (NSP)[b]. Optional.
NSP	parm,[number]	Add a new nonstandard parameter named "parm," and optionally give it value indicated.
NSPn	parm,[number]	Change nth nonstandard parameter and value.
NSPn,X		Delete the nth nonstandard parameter entry.

TABLE A-2. SUMMARY OF SCENARIO PARAMETERS (CONCLUDED).

Code	Value type[a]	Description
		Output Options (OUTP)[b]. See Section IV.D.8.
OUT	y/n	Produce integration OUTput listing file? Default=yes.
PLT	y/n	Produce PLoTs of calculated results? Default=no.
TAB	y/n	Produce TABulations of calculated results? Default=no.
WDY	y/n	Write Derivatives on output listing? Default=no. Used only if OUT=Y.
WRA	y/n	Write reaction RAtes on output listing? Default=no. Used only if OUT=Y.
WRM	y/n	Model listing in integration output file? Default=no.
SPO	y/n	Spool output files to system printer? Default=no.
PS	mod.spc	Add model species (Plot Species) to plot or tabulate.
PSn	mod.spc	Modify the nth Plot Species.
PSn,X		Delete the nth Plot Species.

[a]Types of legal values for these parameters are as indicated by the following codes:

code = One of several codes appropriate for this parameter.
clock = Clock time, in format DHHMM. D=0 for day 1.
nbr/sch = Either a number (units indicated) or the code "SCHED."
mod.spc = Model species name.
number = A single real number.
model = Model file name.
parm = Scenario parameter name.
p.cmp = Pollutant composition file name. Concentrations in ppm.
e.cmp = ROG emissions composition file name. Should be normalized.
sp/ecmp = ROG species name or emissions composition file name.
value = Parameter value. Depends on type of varied parameter.
y/n = Yes or No.

[b]Parameter group code as used by the LIST command.

TABLE A-3. SUMMARY OF SCEN MODE USER INTERFACE COMMANDS.

Command	Explanation
parameter[=value]	Changes the parameter to the new value.
DELETE	Deletes saved scenario.
ENTER	Ask for values of all scenario parameters.
EXPL	Lists the explanatory text file.
EXIT	Return to MAIN mode. Scenario still in memory.
HELP [category]	Gives help display. (HELP HELP lists categories).
INCOM	Lists undefined scenario parameters.
LIST option	List parameters for scenario in memory. Options are ALL, SCEN, PHYS, CHEM, VP, NSP, OUTP, SCHEDS.
LIST SCHED parm	Display the schedule for the specified parameter.
LIST ALLCOMP	List all available composition filenames.
LIST ALLSCE	List all available scenario names.
NEW	Deletes current scenario in memory.
QUIT	Return to MAIN mode. Scenario still in memory.
SAVE	Save scenario and return to MAIN mode.
SCHED parm	Enter mode to edit schedule for named parameter. Schedule commands are as follows:

Command	Explanation
LIST	Lists the schedule and all schedule parameters.
TY=code	Give schedule type. L=linear, S=stepped.
IF=number	Give initial fraction (emissions schedules).
time=number	Give parameter value at clock time.
time=X	Deletes the schedule entry for the time given.
NEW	Deletes all entries in the schedule.
<cr>, Q, or QUIT	Returns to SCEN mode.

TABLE A-4. SUMMARY OF COMP MODE USER INTERFACE COMMANDS.

Command	Explanation
name number	Enters an amount for the named species.
+name number	Increase the amount for the named species by the amount entered.
name,X	Deletes the named species from the file.
ANNOT	List all legal species names with descriptions.
DELETE	Deletes current composition file from memory and disk.
FIND string	List all legal species whose names contain substring entered.
HELP [category]	Gives help display. (HELP HELP lists categories).
LIST	Lists contents of current composition file.
NORM	Normalizes the species levels in the current file (ROG emissions files only).
NEW	Edit new composition file.
QUIT	Exits COMP mode and returns to MAIN.
SAVE	Saves current composition file.
SPEC	List all legal species names and (for ROG species) number of carbons.

TABLE A-5. SUMMARY OF PLOT OPTION COMMANDS.

Command	Explanation
<cr>	Exits the OPTION level and enters the CURVE or DATA level.
IN=name	Takes the Plot command input from file "name.PLT" (VAX systems) or "PLTname" (Cyber systems).
OUT=name	Plot output goes to a file "name.PLO" (VAX systems) or "PLOname" (Cyber systems).
OUT=CRT	Plot output to user terminal (default).
PPP=n	Number of plots per page (for file output only).
TABL	Produce tabulations rather than plots.
NOTA	Produce plots rather than tabulations (default).
CLOCK	Time axis label on the plots given as clock time (HHMM).
HELP (or HE)	Gives a display of the available plot options.
HOUR	Time axis label given as hours after midnight.
MINU	Time axis label given as minutes after midnight (default).
XMAX=minutes	Gives the maximum time for which data are plotted or tabulated. Must be given as minutes after midnight.
XMIN=minutes	Gives the minimum time for which data are plotted or tabulated. Must be given as minutes after midnight.
NOXMAX	Cancels both the XMAX and the XMIN option.
NOXMIN	Same as NOXMAX.
QUIT (or Q)	Exit plot mode.

TABLE A-6. LIST OF NAMES OF SPECIES OR CLASSES WHICH CAN BE INCLUDED IN COMPOSITION FILES FOR ROG EMISSIONS.

	Unreactive compounds				
INERT					
	Oxygenates and Heteroatom-Containing Organics				
FORMALD	ACETALD	PROPALD	ACROLEIN	ACETONE	MEK
GLYOXAL	MEGLYOX	PHENOL	CRESOL	BENZALD	NITROPHE
FURAN	THIOPHEN	PYRROLE			
	n-Alkanes				
N-C4	N-C5	N-C6	N-C7	N-C8	N-C9
N-C10	N-C11	N-C12	N-C13	N-C14	N-C15
	Individual Branched and Cyclic Alkanes				
ISO-C4	ISO-C5	NEO-C5	2-ME-C5	3-ME-C5	23-DMB
CYC-C6	3-ME-C6	4-ME-C6	24-DM-C5	23-DM-C5	ME-CYCC6
4-ME-C7	ISO-C8	ET-CYCC6	4-ET-C7	4-PR-C7	
	Unspeciated Branched Alkanes				
BR-C6	BR-C7	BR-C8	BR-C9	BR-C10	BR-C11
BR-C12	BR-C13	BR-C14	BR-C15		
	Unspeciated Cyclic Alkanes				
CYC-C7	CYC-C8	CYC-C9	CYC-C10	CYC-C11	CYC-C12
CYC-C13	CYC-C14	CYC-C15			
	Benzene and Alkylbenzenes				
BENZENE	TOLUENE	M-XYLENE	O-XYLENE	P-XYLENE	123-TMB
124-TMB	135-TMB				
	Unspeciated Alkylbenzenes				
ALK1BENZ	ALK2BENZ	ALK3BENZ	ALK5BENZ	ALK6BENZ	
	Tetralin and Naphthalenes				
TETRALIN	NAPHTHAL	ME-NAPH	DM-NAPH		
	Alkenes				
ETHENE	PROPENE	1-BUTENE	C-2-BUTE	T-2-BUTE	ISOBUTEN
1-PENTEN	2M-1-BUT	2M-2-BUT	1-HEXENE	23M2-BUT	
	Unspeciated External Alkenes				
C6-OLE1	C7-OLE1	C8-OLE1	C9-OLE1	C10-OLE1	C11-OLE1
C12-OLE1	C13-OLE1	C14-OLE1	C15-OLE1		
	Unspeciated Internal Alkenes				
C6-OLE2	C7-OLE2	C8-OLE2	C9-OLE2	C10-OLE2	C11-OLE2
C12-OLE2	C13-OLE2	C14-OLE2	C15-OLE2		

TABLE A-7. LIST OF NAMES OF INORGANIC AND NONEMITTED ORGANIC POLLUTANTS WHICH CAN BE INCLUDED IN COMPOSITION FILES FOR BACKGROUND OR ALOFT SPECIES.

Model name	Description
	Inorganic Species
O3	Ozone
NO	Nitric Oxide
NO2	Nitrogen Dioxide
HONO	Nitrous Acid
HONO2	Nitric Acid
HO2H	Hydrogen Peroxide
N2O5	Nitrogen Pentoxide
NO3	Nitrate Radicals
HO2NO2	Peroxy Nitric Acid
	Nonemitted Organic Products and Intermediates
C-PAN	Peroxyacetyl Nitrate (PAN)
C2-PAN	Peroxyproponyl Nitrate and other PAN analogues
BZ-PAN	Peroxy Benzoyl Nitrate
HCO-PAN	PAN analogue formed from glyoxal
RONO2	Alkyl Nitrates
NITROPHEN	Nitrophenols

APPENDIX B

INSTALLATION MANUAL AND DATA FILES REQUIRED

In this Appendix, instructions are given on how to install this model on DEC VAX systems running under the VMS operating system, and on CDC Cyber systems running under NOS. This Appendix also gives a list of all the files required for the implementation of the initial version of the model. The specific programs included with this model are the following:

- The User Interface program
- The Model Integration program
- The Plotting program
- The Model Preparation program
- The ALKCONV program (Used for model modifications. Not documented in this manual)

The specific types of files included are: (1) the source code and compile and linking command files required to install these programs; (2) the input files required to install the models discussed in Section III of this report; (3) the data files required to install the scenarios, composition files, and to run the example calculations discussed in Section V; and (4) the data files required to make model modifications as discussed in Appendix C of this report.

In addition to the files required to install this model and run the examples as discussed in this report, the initial distribution of this model also includes text files giving more detailed documentation of the Integration, Preparation, and Plotting programs than discussed in this volume. It should be recognized, however, that the documentation in these text files are still in draft form, since there was insufficient time or funds in this program to comprehensively document aspects of the software not required for routine use of this model. This draft documentation may be revised and expanded in the future if funds become available for this purpose. However, the information given there should be useful to serve as a preliminary technical reference guide for the capabilities of this software, including discussion of features not utilized when using this

model with the User Interface. Persons interested in more comprehensive documentation of these features should contact the authors of this manual.

The files required for the installation of this model are available on two computer tapes: one with the files required for installation on VAX systems, and one with the files for installation on Cybers. However, the discussion in this Appendix is based on the assumption that the necessary files have already been copied to a disk on the computer on which the model is to be installed; the specific instructions on how to copy the files from the tape to disk depends on how the tape was created, and will be included with the tapes when they are distributed. This model has already been installed on the VAX computer at Brooks AFB and the Cyber computer at Tyndall AFB, so it is not necessary to go through this installation procedure to utilize this model at those facilities.

A. INSTALLATION AND FILES ON VAX SYSTEMS

1. List of Files and Recommended Directory Structure

The data files included in the distribution of this model for VAX systems are listed in Table B-1. This listing also includes the files which are created in the process of installing the model or running the example calculations discussed in Section V. A brief description of these files is also included in the table.

It is recommended that when installed on VAX systems, the files be organized according to the directory structure indicated in Figure B-1. The listing in Table B-1 indicates the directories into which each of the files should be placed. The installation command files provided assume that this directory structure is utilized. This structure has all of the source code and compile and link command files in the ".SOURCE" subdirectory, with subdirectories of that containing the source code specific to the individual programs; has the model installation data files (such as, for example the files giving the chemical mechanism and other Preparation program input files) in the ".MODEL" subdirectory; and the files used by the User Interface, and maintained by the user, in the "root" directory, which is the default directory for routine use of this

TABLE B-1. LIST OF FILES INCLUDED IN THE INITIAL INSTALLATION OF THE MODEL ON VAX SYSTEMS.

A. Files in the ["Root"] Directory

1. Data Files Required and Maintained by the User Interface.

ALLSCEN.LIS	Contains scenario names.
ALLCOMP.LIS	Contains composition file names.
name.SCP	Contains data defining scenarios, where "name" is the scenario name. The initial distribution includes the following .SCP files, but more can be created by the user.

EKMA1.SCP EKMA2.SCP EKMA3.SCP
EKM1A.SCP EKM1B.SCP

name.CMP	Contains composition file data, where "name" is the composition file name. The initial distribution contains the following .CMP files, but more can be created by the user.

FUEL1.CMP FUEL2.CMP FUEL3.CMP
FUEL1F.CMP FUEL1T.CMP FUEL1P.CMP
EKMAALO.CMP SAISURG.CMP SYNEXH.CMP

SUBMIT.PRM	Contains highest calculation number prepared by the CALC command of the User Interface program.
SPECIES.LIS	Contains list of names of species which can be included in composition files. Also gives the number of carbons for emitted ROG species and comments describing the types of chemicals each species is used to represent. Note that the data in this file is model dependent, and that adding new species to this model requires more than just adding new names to this file.
MODELS.LIS	Contains model names which can be used with these files. If any models are to be added, the name of the new model must be added to this file using the system editor.
HELP.TXT	Text file of HELP messages for the User Interface program. This file can be edited to modify the help messages.

TABLE B-1. LIST OF FILES INCLUDED IN THE INITIAL INSTALLATION OF THE MODEL ON VAX SYSTEMS (CONTINUED).

EXPLAIN.TXT	Explanatory text about the User Interface program. This file can be edited to modify the explanatory text output when the MAIN mode EXPL command is given.

2. Data Files Maintained or Produced by the Integration or Plotting Programs

CALCnnnn.CDT	Binary output file from the Integration program containing calculated concentration-time data from calculation number nnnn. Used as input to the Plotting program or the PLOT mode of the User Interface. The initial distribution contains at least 64 such files, consisting of results of the calculations in the examples discussed in Section V.B.
CALCnnnn.OUT	Integration program listing output file for calculation group nnnn, where nnnn is the calculation number for the last calculation produced as a result of a single User Interface CALC command. The initial distribution will contain at least one such file created by one of the example calculations discussed in Section V.B, but will not contain them from all of the example calculations because of the amount of disk space they require.
CALCnnnn.PLO	Output file from Plotting program for calculation group nnnn, where nnnn is the calculation number for the last calculation produced as a result of a single User Interface CALC command. The initial distribution contains at least 16 such files, consisting of results of the calculations in the examples discussed in Section V.B.
CALC.LOG	Log of all calculations run. Consists of a list of the calculation number, label, model name, and date for all calculations run. In the initial distribution, it will contain entries for the example calculations discussed in Section V.B, and for any test calculations carried out when installing the model.
CALC.PRM	Contains calculation number of last calculation processed by the Integration program.

TABLE B-1. LIST OF FILES INCLUDED IN THE INITIAL INSTALLATION OF THE MODEL ON VAX SYSTEMS (CONTINUED).

3. Temporary Files Created by the User Interface For Calculations Which Are Deleted When Calculations are Completed. (These files are not included in the initial distribution.)

AFJOB.COM	Batch command file for calculations created by CALC commands during a session at the User Interface. Deleted when the calculations are completed, if run real time. If calculations are submitted for batch processing, the user should delete this file when processing is completed.
CALCnnnn.INT	Integration program input file for calculation group nnnn, where nnnn is the calculation number for the last calculation produced as a result of a single User Interface CALC command.
CALCnnnn.PLT	Plotting program input file for calculation group nnnn, where nnnn is the calculation number for the last calculation produced as a result of a single User Interface CALC command.

4. Command Files

BLDALL.COM	Compiles and links all programs. Not required for routine use of the model.
RUNSCEN.COM	Used to run User Interface program. Includes the commands to automatically submit or execute calculations created by the CALC commands.
PLT.COM	Used to run Plot program (optional). Allows the user to specify the .PLT filename on the command line. I.e., "@PLT file" results in the Plotting program being run with input from "file.PLT," and output going to "file.PLO."

5. Executable Files. (These are not initially provided, but are produced when BLDALL.COM is executed)

SCEN.EXE	The User Interface program
INT.EXE	The Model Integration program
PLOT.EXE	The Plotting program

TABLE B-1. LIST OF FILES INCLUDED IN THE INITIAL INSTALLATION OF THE MODEL ON VAX SYSTEMS (CONTINUED).

---

6. Directory Parameter File. Required by all programs when run from the "root" directory.

MODELING.PRM	Contains names of subdirectories in which various types of files are located

B. Files in the [.MODEL] Directory

1. Preparation Program Input Files for the Distributed Chemical Models.

name.PRP	Main Preparation program input files for each of the chemical "models" which are implemented, where "name" is the model name, as referenced by the MOD scenario parameter. The initial distribution of this model includes four different chemical models based on differing ROG lumping techniques, and the preparation input files for each of these are as follows:

LPARM.PRP LM1ARM.PRP LM2ARM.PRP
LPBRM.PRP

name.RXN	Preparation Program Input files for the reactions of the various chemical species in the model. These are used by the .PRP files listed above. The .RXN files included in the initial distribution are as follows:

Inorganic Reactions. Used in all models.

NOXAIR.RXN

Chamber-dependent reactions used when modeling chamber experiments. Used in all models.

WALLS.RXN

General peroxy radical reactions. Used in all models.

RO2.RXN

Reactions of individual organics or "lumped molecule" species and their products which are used in all models.

TABLE B-1. LIST OF FILES INCLUDED IN THE INITIAL INSTALLATION OF THE MODEL ON VAX SYSTEMS (CONTINUED).

HCHO.RXN	CCHO.RXN	RCHO.RXN
ACETONE.RXN	MEK.RXN	BZALD.RXN
GLYOXAL.RXN	MEGLYOX.RXN	PHENOL.RXN
CRESOL.RXN	RONO2.RXN	ETHENE.RXN
FURAN.RXN	THIOPHEN.RXN	PYRROLE.RXN

Reactions of individual organics or "lumped molecule" species used in models with LM1 or LM2 lumping.

C4C5ALK.RXN	C6PALK.RXN	C9PALK.RXN
PROPENE.RXN	T2BUT.RXN	1HEXENE.RXN
BENZENE.RXN	TOLUENE.RXN	MXYLEN.RXN
TETRALIN.RXN	NAPHTHAL.RXN	MENAPH.RXN
23DMNAPH.RXN		

Rreactions of "lumped molecule" species used in models with LP lumping.

LMPBZC3.RXN	LMPC4C5.RXN	ALKARO1.RXN
ALKARO1.RXN	ALKARO2.RXN	ALKARO3.RXN
ALKARO4.RXN	ALKARO5.RXN	LMPOLE1.RXN
LMPOLE2.RXN		

2. Lumping Parameter Files. These indicate how ROC species not explicitly on the models are to be represented. Read by the Integration program, but must be consistent with the model being used. Three "master" files are included, one for each lumping approach, which must be copied into the files used by each model. The files required by the initially distributed models are as follows.

LM1.LMP	Lumping file for models using LM1 lumping
LM2.LMP	Lumping file for models using LM2 lumping
LP.LMP	Lumping file for models using LP lumping
name.LMP	Lumping parameter file for model "name" which is used to convert background or aloft mix composition file data into absolute concentrations. Specific files provided:

LPBRM.LMP (same as LP.LMP)
LPARM.LMP (same as LM1.LMP)
LM1ARM.LMP (same as LM1.LMP)
LM2ARM.LMP (same as LM2.LMP)

TABLE B-1. LIST OF FILES INCLUDED IN THE INITIAL INSTALLATION OF THE MODEL ON VAX SYSTEMS (CONTINUED).

	name.LMB	Lumping parameter file for model "name" which is used to convert data in composition files giving base case ROG emissions into emission rates and initial concentrations of model species. Specific files provided: LPBRM.LMB (same as LP.LMP) LPARM.LMB (same as LM1.LMP) LM1ARM.LMB (same as LM1.LMP) LM2ARM.LMB (same as LM2.LMP)
	name.LMT	Lumping parameter file for model "name" which is used to convert data in composition files giving test mixture ROG emissions into emission rates and initial concentrations of model species. Specific files provided: LPARM.LMT (same as LP.LMP) LM1ARM.LMT (same as LM1.LMP) LM2ARM.LMT (same as LM2.LMP)
	ALKANE.LPM	Kinetic and mechanistic parameter file for alkanes. Required by models using LP lumping and referenced by LP.LMP and its copies.
	AROMATIC.LPM	Kinetic and mechanistic parameter file for aromatics. Required by models using LP lumping and referenced by LP.LMP and its copies.
	ALKENE.LPM	Kinetic and mechanistic parameter file for alkenes. Required by models using LP lumping and referenced by LP.LMP and its copies.
	ALKANE.PRM	Structural data for the alkanes for which LP lumping is employed. Used as input to the ALKCONV program to produce the ALKANE.LPM file.
3.	name.MOD	Model Data Files, where "name" is the name of the model. Not initially distributed, but produced by executing the preparation program to prepare the distributed models for use. These are required by the Integration program to do calculations using the models. After installation of this model, the following .MOD files will exist.

TABLE B-1. LIST OF FILES INCLUDED IN THE INITIAL INSTALLATION OF THE MODEL ON VAX SYSTEMS (CONTINUED).

	LPARM.MOD	LM1ARM.MOD	LM2ARM.MOD
	LPBRM.MOD		

4. name.PRO — Model Preparation Listing Output Files, where "name" is the name of the model. These are not required by any of the programs, but serve to document the model. It is recommended that they be printed before they are deleted.

LPARM.PRO	LM1ARM.PRO	LM2ARM.PRO
LPBRM.PRO		

5. Executable Files. (Not initially distributed, but produced by executing the BLDALL.COM command file.)

PREP.EXE	The Model Preparation program
ALKCONV.EXE	The "Alkane" program, which converts alkane structural parameters in ALKANE.PRM to produce the mechanistic and kinetic parameters in ALKANE.LPM.

6. Directory Parameter File. Required by the Preparation program.

MODELING.PRM	Contains names of subdirectories into which various types of files are located.

C. Files in the [.SOURCE] Directory

1. FORTRAN Source Code

NEWSUBS.FOR	Utility subroutines used by most programs.
NEWINBUF.FOR	Utility subroutines controlling terminal or file input. Used by most programs.
NEWIOBUF.FOR	"Include" file containing specifications of data used by NEWINBUF subroutines, and I/O buffers and miscellaneous specifications.

2. NEWSUBSCL.COM — Compiles NEWSUBS.FOR and NEWINBUF.FOR, and places the object files in the object library NEWSUBS.OLB. Used by BLDALL.COM.

TABLE B-1. LIST OF FILES INCLUDED IN THE INITIAL INSTALLATION OF THE MODEL ON VAX SYSTEMS (CONTINUED).

3. NEWSUBS.OLB — Object files for NEWSUBS and NEWINBUF. Not initially distributed. Created by NEWSUBSCL.COM.

D. Files in the [.SOURCE.SCEN] Subdirectory

1. FORTRAN Source Files for Modules of the User Interface Program. These are as follows:

MAIN.FOR	EDSC.FOR	INTFIL.FOR	PLTFIL.FOR
CALC.FOR	HELPSB.FOR	LIST.FOR	SETSCN.FOR
COMP.FOR	INPSCN.FOR	ONEBY1.FOR	INBUF6.FOR
DECLIN.FOR	INPUT.FOR	PLOTSB.FOR	

2. AFSPECS.FOR — "Include" file containing the specifications for all the COMMON variables used in the User Interface program. Referenced in most User Interface source files.

3. SCENCL.COM — Command file to compile and link the User Interface program. Executable file is in the [.SOURCE.SCEN] directory. Must be copied to the ["root"] directory before use.

E. Files in the [.SOURCE.PLOT] Subdirectory

1. FORTRAN Source Files for Modules of the Plotting Program. These are as follows:

PLOT.FOR	HELPPL.FOR	PGPLT.FOR	CHADAT.FOR
CALDAT.FOR	TEKPLT.FOR	DMPDAT.FOR	RATDAT.FOR
DASHSYM.FOR	GENDAT.FOR	TABUL.FOR	LCHAR.FOR
TEKDUM.FOR			

2. PLTSPECS.FOR — "Include" file containing the specifications for all the COMMON variables used in the Plotting program. Referenced in most Plotting program source files.

3. PLTCL.COM — Command file to compile and link the Plotting program. Executable file is in the [.SOURCE.PLOT] directory. Must be copied to the ["root"] directory before use.

TABLE B-1. LIST OF FILES INCLUDED IN THE INITIAL INSTALLATION OF THE MODEL ON VAX SYSTEMS (CONTINUED).

---

4. PLOT.DOC — Draft documentation of the features of the plotting program, including features not documented in this manual.

F. Files in the [.SOURCE.INT] Subdirectory

1. FORTRAN Source Files for Modules of the Integration Program. These are as follows:

INTMAIN.FOR	DUMOPT.FOR	LUMPHC.FOR	PHOTKS.FOR
RDPHK.FOR	GETCAL.FOR	MODL.FOR	JMATRX.FOR
RDVCO.FOR	INT.FOR	NEWPHK.FOR	PRNTC.FOR
SDREAD.FOR	INTINI.FOR	NEWRK.FOR	RDCKC.FOR
STOREC.FOR	NEWSTP.FOR	RDCMP.FOR	WRLMO.FOR
INTPRP.FOR	PFALIN.FOR	RDLMP.FOR	XQKODE.FOR
LSODE.FOR			

2. MODSUB.MAR — Assembly language source file for VAX specific portion of Integration program, which permits a single executable program to run all models. Required for execution of model-specific subroutines produced by the Preparation program. (An all-Fortran version of the Integration program which runs on VAX systems is available from the authors upon request. This version produces model-specific Integration programs, as is the case on Cyber systems).

3. MAINSPC.FOR — "Include" file containing the specifications for all the COMMON variables used in the Integration program, except for those used by the lumping subroutines. Referenced in most Integration program source files.

4. LMPSPECS.FOR — "Include" file containing specifications of the lumping subroutines. Referenced by RDLMP, RDCMP, WRLMO, and LUMPHC.

5. INTCL.COM — Command file to compile and link the Integration program. Executable file is in the [.SOURCE.INT] directory. Must be copied to the ["root"] directory before use.

TABLE B-1. LIST OF FILES INCLUDED IN THE INITIAL INSTALLATION OF THE MODEL ON VAX SYSTEMS (CONCLUDED).

6.	INT.DOC	Draft documentation of the features and usage of the Integration program, which are not documented in this manual.

G. Files in the [.SOURCE.PREP] Subdirectory

1. FORTRAN Source Files for Modules of the Preparation Program. These are as follows:

PREP.FOR	ASMD.FOR	LISTSC.FOR	RDRXN.FOR
BLDUP.FOR	PNHRXN.FOR	REORDR.FOR	DIFFUN.FOR
RXLST1.FOR	SAVERX.FOR		

2.	PSPECS.FOR	"Include" file containing the specifications for all the COMMON variables used in the Preparation program. Referenced in most Preparation program source files.
3.	PRPCL.COM	Command file to compile and link the Preparation program. Executable file is in the [.SOURCE.PREP] directory. Must be copied to the [.MODEL] directory before use.
4.	PREP.DOC	Draft documentation of the features and usage of the Preparation program, which are not documented in this manual.

H. Files in the [.SOURCE.ALKANE] Subdirectory

1. FORTRAN Source Files for Modules of the ALKCONV Program. These are as follows:

ALKCONV.FOR	NEWSTR.FOR	ALKANE.FOR	SUMTYP.FOR
ISOM.FOR			

2.	ALKSPECS.FOR	"Include" file containing the specifications for all the COMMON variables used in the ALKCONV program. Referenced in most ALKCONV program source files.
3.	ALKCNVCL.COM	Command file to compile and link the ALKCONV program. Executable file is in the [.SOURCE.ALKANE] directory. Must be copied to the [.MODEL] directory before use.

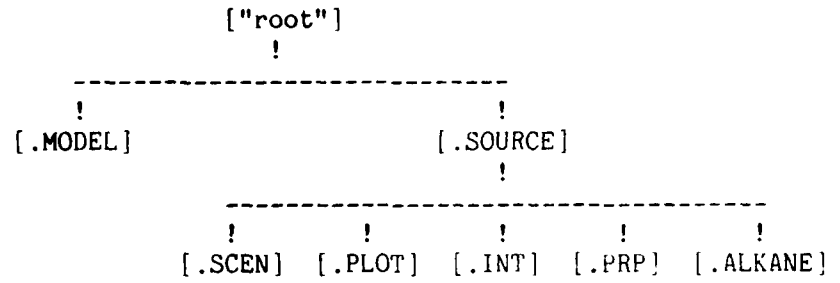


Figure B-1. Directory "Tree" Structure for Installation of the Model on VAX Systems.

model. All these directories should be created and the appropriate files should be copied into them before executing any of the installation command files. It is assumed that the installer of this model knows the system commands necessary to accomplish this.

Note that the "root" directory for this model on VAX systems does not necessarily have to be a log-in root directory, only that it has to be the top level directory to be used for the files of this modeling system. In general, once installed, the user need not be further concerned with the subdirectories. The User Interface program, (SCEN), the Integration program, (INT), and the Plotting program, (PLOT), will all be run from the root directory. Model modification using the Model Preparation program, (PREP), and the ALKCONV program will be done in the [.MODEL] subdirectory.

2. Compilation of Programs Required by the Model

a. Compilation of All Programs

A command file, called BLDALL.COM, is provided to install all of the executable programs required for the installation and utilization of this model. BLDALL.COM actually runs each of the compile and link command files in the source subdirectories, and then moves the executable files to the ["root"] or [.MODEL] directories, as appropriate. Before this command file can be executed, all the source and command files must

be in the subdirectories as indicated in Table B-1. To run this command file, set the default directory to the "root" for this model, and then enter:

```
@BLDALL
```

The text of the BLDALL command file is as follows:

```
$ SET VERIFY
$ SET DEFAULT [.SOURCE]
$ @NEWSUBSCL
$ !
$ SET DEFAULT [.PLOT]
$ @PLTCL
$ RENAME PLOT.EXE [.-.-]PLOT.EXE
$ !
$ SET DEFAULT [.-.SCEN]
$ @SCENCL
$ RENAME SCEN.EXE [.-.-]SCEN.EXE
$ !
$ SET DEFAULT [.-.INT]
$ @INTCL
$ RENAME INT.EXE [.-.-]INT.EXE
$ !
$ SET DEFAULT [.-.PRP]
$ @PRPCL
$ RENAME PREP.EXE [.-.-.MODEL]PREP.EXE
$ !
$ SET DEFAULT [.-.ALKANE]
$ @ALKCNVCL
$ RENAME ALKCONV.EXE [.-.-.MODEL]ALKCONV.EXE
$ !
$ SET NOVERIFY
$ SET MESSAGE/IDENT/TEXT/FACIL/SEVER
$ SET DEFAULT [.-.-]
```

Note that the compile/link command files all use the /LIST option for compiling, which will generate listing files for all the source files. These can be used for debugging, if necessary. It is recommended, however, that the listing files in each of the subdirectories be deleted after the programs have been linked and tested, to save disk space. The object files can also be deleted after the successful installation and testing of this model.

The programs can also be compiled and linked individually by running the command files for them in their subdirectories. This is discussed below for the four programs. In all cases the command file places the executable file it creates in the source directory for the program; these need to be moved either to the "root" or the [.MODEL] directory, as appropriate, before they can be used routinely. However, if the program is being modified, it may be desirable to leave the executable file for the program in the source directory while it is being tested and debugged before copying it to the directories for the working versions of the programs.

b. Compiling the Utility Subroutines

Before any of the programs can be linked, it is necessary to compile the utility subroutines which they use, whose source files are in the [.SOURCE] directory, to create the library of their object code. To do this, set the default directory as [.SOURCE], and then give the command

```
@NEWSUBSCL
```

This causes the following commands to be executed:

```
$ SET MESSAGE/NOIDENT/NOSEVER/NOTEXT/NOFACIL
$ DEL NEWINBUF.OBJ;*,NEWINBUF.LIS;*,NEWSUBS.OBJ;*/NOCONFIRM
$ DEL NEWSUBS.LIS;*/NOCONFIRM
$ DEL NEWSUBS.OLB;*/NOCONFIRM
$ SET MESSAGE/IDENT/SEVER/TEXT/FACIL
$ FOR/NOI4/LIST NEWINBUF,NEWSUBS
$ LIB/CREATE NEWSUBS.OLB NEWINBUF.OBJ,NEWSUBS.OBJ
```

After this command file is executed, the object and listing files can be deleted. However, the NEWSUBS.OLB file should be retained because it is required when linking the programs used in this model.

c. Compiling and Linking the Plot Program

The Plotting program is the first program linked when running the BLDALL.COM file because some of its subroutines are required before the User Interface program can be linked. To create the Plotting program, set the [.SOURCE.PLOT] subdirectory as the default, and give the command

```
@PLOTCL
```

This executes the PLOTCL.COM file, whose contents are listed below:

```
$ SET MESSAGE/NOIDENT/NOSEVER/NOTEXT/NOFACIL
$ DEL *.OBJ;*, *.LIS;*/NOCONFIRM
$ SET MESSAGE/IDENT/SEVER/TEXT/FACIL
$ !
$ FOR/NOI4/LIST PLOT,CALDAT,CHADAT,DMPDAT,GENDAT
$ FOR/NOI4/LIST HELPPL,PGPLT,TABUL,RATDAT
$ !
$ !  IF PLOT10 LIBRARY IS AVAILABLE, USE THE FOLLOWING TO LINK THE PLOT
$ !  PROGRAM, ADDING FIRST THE APPROPRIATE DIRECTORY FOR THE PLOT10
$ !  LIBRARY.
!
!$ FOR/NOI4/LIST LCHAR,TEKPLT,DASHSYM
!$ LINK PLOT,CALDAT,CHADAT,DMPDAT,GENDAT,HELPPL, -
!    PGPLT,TABUL,RATDAT,TEKPLT,DASHSYM,LCHAR,[.-]NEWSUBS/LIBR, -
!    PLOT10-LIB/LIBR
$ !
$ !  IF PLOT10 LIBRARY IS NOT AVAILABLE, USE THE FOLLOWING TO LINK THE
$ !  PLOT PROGRAM
$ !
$ FOR/NOI4/LIST TEKDUM
$ LINK PLOT,CALDAT,CHADAT,DMPDAT,GENDAT,HELPPL, -
   PGPLT,TEKDUM,TABUL,RATDAT,[.-]NEWSUBS/LIBR
```

The listing files created by this command can be deleted, but the object files should not be deleted until the User Interface program has been successfully linked.

Note that if the PLOT10 library is available on the computer on which this modeling system is installed, you may wish to edit the PLTCL command file to create a version of the Plotting program which can use the PLOT10 capabilities. The command file, as provided, will create a version of the Plotting program with capabilities only to create page-plots and

tables, (without using the PLOT10 library). But also included in the command file, as comments, are lines in the general form for linking to the PLOT10 library. If this is to be used, the appropriate directory and name for the PLOT10 library on your computer system will have to be added. However, the use of the PLOT10 capabilities requires special input to the program, which is not presently documented.

After the Plotting program has been successfully linked and (if modified) tested, it should be moved to the "root" directory where it is available for normal use. This can be done by the command

```
RENAME PLOT.EXE [.-.-]PLOT.EXE
```

The old PLOT.EXE in the "root" directory should be deleted first.

d. Compiling and Linking the User Interface (SCEN) Program

Before the User Interface program can be linked, a number of the subroutines of the Plotting program must be compiled, and their object files must exist in the [.SOURCE.PLOT] directory. Assuming that this is the case, to compile and link the User Interface program, first set the default directory to [.SOURCE.SCEN], and then give the command

```
@SCENCL
```

This causes the following commands to be executed:

```
$ SET MESSAGE/NOIDENT/NOSEVER/NOTEXT/NOFACIL
$ DEL *.OBJ;*/NOCONFIRM
$ DEL *.LIS;*/NOCONFIRM
$ SET MESSAGE/IDENT/SEVER/TEXT/FACIL
$ !
$FOR/NOI4 MAIN,SETSCN,EDSC,INPSCN,ONEBY1,INPUT,LIST,HELPSB
$FOR/NOI4 COMP,PLTFIL,CALC,INTFIL,DECLIN,PLOTSB,INBUF6
$LINK MAIN,SETSCN,EDSC,INPSCN,ONEBY1,INPUT,LIST,HELPSB, -
   COMP,PLTFIL,CALC,INTFIL,DECLIN,PLOTSB,INBUF6, -
   [.-.PLOT]PGPLT,TABUL,CHADAT,CALDAT,GENDAT,DMPDAT, -
   [.-]NEWSUBS.OLB/LIB
$RENAME MAIN.EXE SCEN.EXE
```

The listing and object files can be deleted after the successful conclusion of these commands. If no longer needed, the Plotting program object files in [.SOURCE.PLOT] can be deleted also.

After the SCEN program has been successfully linked and (if modified) tested, it should be moved to the "root" directory where it is available for normal use. This can be done by the command

```
RENAME SCEN.EXE [.-.-]SCEN.EXE
```

The old SCEN.EXE in the "root" directory should be deleted first.

e. Compiling and Linking the Model Integration Program

To compile and link the Integration program, first set the default directory to be [.SOURCE.INT], and then enter the command

```
@INTCL
```

This causes the following commands to be executed:

```
$ SET MESSAGE/NOIDENT/NOSEVER/NOTEXT/NOFACIL
$ DEL *.OBJ;*,*.LIS;*,INT.OLB;*/NOCONFIRM
$ SET MESSAGE/IDENT/SEVER/TEXT/FACIL
$ MACRO/LIST MODSUB
$ FOR/NOI4/LIST INTMAIN,XQKODE,GETCAL,INT,MODL,RDVCO
$ FOR/NOI4/LIST PHOTKS,PFALIN,PRNTC,RDCKC,SDREAD,STOREC
$ FOR/NOI4/LIST INTINI,INTPRP,NEWSTP,RDPHK,JMATRX,NEWPHK,NEWRK
$ FOR/NOI4/LIST DUMOPT
$ FOR/NOI4/LIST LSODE
$ FOR/NOI4/LIST WRLMO,LUMPHC,RDCMP,RDLMP
$ !
$ LIBR INT.OLB/CREATE
$ LIBR INT.OLB XQKODE.OBJ,GETCAL.OBJ,INT.OBJ,MODL.OBJ/INSERT
$ LIBR INT.OLB PHOTKS.OBJ,PFALIN.OBJ,PRNTC.OBJ,RDCKC.OBJ/INSERT
$ LIBR INT.OLB LSODE.OBJ,JMATRX.OBJ,STOREC.OBJ,MODSUB.OBJ/INSERT
$ LIBR INT.OLB INTINI.OBJ,INTPRP.OBJ,SDREAD.OBJ/INSERT
$ LIBR INT.OLB NEWSTP.OBJ,RDPHK.OBJ,NEWRK.OBJ,NEWPHK.OBJ/INSERT
$ LIBR INT.OLB DUMOPT.OBJ,RDVCO.OBJ/INSERT
$ LIBR INT.OLB WRLMO.OBJ,LUMPHC.OBJ,RDLMP.OBJ,RDCMP.OBJ/INSERT
$ !
$ LINK INTMAIN,INT.OLB/LIB,[.-]NEWSUBS.OLB/LIB
$ RENAME INTMAIN.EXE INT.EXE
```

Note that once all the object files are in the library INT.OLB, they can be deleted, even if it may be necessary to recompile some of the subroutines. The INT.OLB object library file can be deleted after the successful linking of this program.

After the INT program has been successfully linked and (if modified) tested, it should be moved to the "root" directory where it is available for normal use. This can be done by the command

```
RENAME INT.EXE [.-.-]INT.EXE
```

The old INT.EXE in the "root" directory should be deleted first.

f. Compiling and Linking the Model Preparation Program

To create the Model Preparation program, set the default directory to [.SOURCE.PRP], and then give the command

```
@PRPCL
```

This will cause the following commands to be executed.

```
$ SET MESSAGE/NOIDENT/NOSEVER/NOTEXT/NOFACIL
$ DEL *.LIS;*/NOCONFIRM
$ DEL *.OBJ;*/NOCONFIRM
$ SET MESSAGE/IDENT/SEVER/TEXT/FACIL
$ FOR/NOI4/LIST ASMD,PNHRXN,SAVERX
$ FOR/NOI4/LIST PREP,DIFFUN,LISTSC,RDRXN,RXLST1
$ FOR/NOI4/LIST VCOEF,BLDUP,REORDR
$ LINK PREP,ASMD,DIFFUN,LISTSC,PNHRXN,RDRXN,RXLST1, -
   SAVERX,VCOEF,BLDUP,REORDR,[.-]NEWSUBS.OLB/LIB
```

The object and listing files can be deleted after successful execution of these commands.

After the PREP program has been successfully linked and (if modified) tested, it should be moved to the [.MODEL] directory where it is available to prepare the distributed models. This can be done by the command

RENAME PREP.EXE [.-.-.MODEL]PREP.EXE

The old PREP.EXE in the [.MODEL] directory should be deleted first.

g. Compiling and Linking the Alkane (ALKCONV) Program

The ALKCONV program is not required to install the initial version of this model, but it is required if it is desired to add additional alkanes which can be lumped using the "LP" approach. To create this program, set the default directory to [.SOURCE.ALKANE], and then give the command

@ALKCNVCL

This causes the following commands to be executed:

```
$ SET MESSAGE/NOIDENT/NOSEVER/NOTEXT/NOFACIL
$ DEL *.LIS;*/NOCONFIRM
$ DEL *.OBJ;*/NOCONFIRM
$ SET MESSAGE/IDENT/SEVER/TEXT/FACIL
$ FOR/NOI4/LIST ALKCONV,ALKANE,NEWSTR,ISOM,SUMTYP
$ LINK ALKCONV,ALKANE,ISOM,NEWSTR,SUMTYP,-
  [.-]NEWSUBS.OLB/LIB
```

After the ALKCONV program has been successfully linked and (if modified) tested, it should be moved to the [.MODEL] directory where it can be used when modifying the model. This can be done by the command

RENAME ALKCONV.EXE [.-.-.MODEL]ALKCONV.EXE

The old ALKCONV.EXE in the [.MODEL] directory should be deleted first.

3. Preparation of Distributed Models

Before any simulations with a model can be carried out, the model must be "prepared," i.e., the Model Preparation program must be run for that model. The initial distribution of this modeling system includes

four models, called LPARM, LPBRM, LM1ARM, and LM2ARM, which differ in how ROG species are lumped. It is necessary to prepare only those models which are actually going to be used, but if a model is not to be prepared, its name should be removed from the MODELS.LIS data set, to prevent non-prepared models being referenced in scenarios created using the User Interface. As indicated in Table B-1, the primary input files for these models are the .PRP files, though the .RXN files must also be present in the same directory, since these are referenced by the .PRP files.

Assuming that the necessary .PRP and .RXN files are available in the [.MODEL] directory, and the Preparation program has been compiled, linked, and moved to the [.MODEL] directory, to prepare a model, first set [.MODEL] as the default directory, and then give the command

```
RUN PREP
```

The program will then give the following prompt:

```
NAME FOR INPUT/OUTPUT FILES:
```

Enter a model name in response to this prompt, i.e., the first part of the name of the .PRP file which is the model preparation input file. For the initial distribution, this can be either LM1ARM, LM2ARM, LPARM, or LPBRM. The program will then take the rest of its input from the appropriate .PRP file and the reaction description files (.RXN files). Upon completion, the PREP program will give the message:

```
MODEL='name      ' PREPARED
```

where "name" is the model name. The file "name.PRO" will have been created, and will give a listing of the model, and any error or warning messages that might have been produced due to incorrect input data. The user can print this file and then delete it if desired. The file "name.MOD" will be created if the preparation is successful. This file is required to run the model, and should be kept as long as the model is to be used. It is a binary file, so it should not be printed.

4. The MODELING.PRM File

When installed on VAX computers, the programs in this modeling system can optionally use a parameter file of directory names, called MODELING.PRM, to determine where to get certain types of input files and where to put certain types of output files. If this parameter file is not present in the directory in which the program is being executed, the program will expect all input files to be in that same directory, and will put all output files in that same directory. However, by using the MODELING.PRM file, different types of data files can be kept in different directories (or subdirectories), and can be kept separate from the source files and executable files, if desired. If this is desired, a MODELING.PRM file must exist on every directory in which these programs might be run.

If this model is to be installed using the recommended directory structure, with the files located as indicated in Table B-1, then the MODELING.PRM files distributed with this model can be used without modification. However, if a different directory structure is to be used, or if the users of the model wish to have calculation output (.CDT) files go into a different directory than the "root" directory, the installer will need to modify this file.

The MODELING.PRM file must be created by the user with an editor, and should have the following format:

```
MOD=[mod-directory]
PHF=[photolysis-directory]
LOG=[log-directory]
TMP=[temporary-directory]
SDR=[spectral-dist-directory]
CAL=[calculated-data-directory]
GDT=[general-data-directory]
IMG=[chamber-data-directory]
LMP=[lumping-data-directory]
```

where each of the "-directory" names is the name of an existing directory, (e.g., MOD=[CHAMBER.MODELS]). These lines may be in any order in the file, and any of them may be omitted. If they are omitted, then the default directory is associated with the missing code. The three-letter code and equal sign must be the first four characters on the line in order to be recognized by the programs. Any line in the file that does not begin with this format will be ignored. Therefore, comment lines or blank lines may be included in the file, if desired. However, for compatibility with possible future versions of this software, it is recommended that any comment lines included in this file begin with a "!" character.

Each of the three programs in this modeling system uses a different set of the above directory codes. The types of files referenced by each of the directory codes, and the programs which utilize them, are summarized in Table B-2.

If all programs are to be executed from the same directory, then all the programs will use the same MODELING.PRM file. If, however, different programs will be executed from different directories, then it is very important that the relevent directory codes match up in the MODELING.PRM files in these different directories. For example, the MOD directory must be the same for the Preparation program as for the Integration program, in order for the Integration program to find the MOD file produced by the Preparation program. Likewise, the CAL directory must be the same for the Plotting program as for the Integration program, in order for the Plotting program to find the .CDT and .RDT files produced by the Integration program. Generally, the LOG directory should be set to be the directory in which the program is being executed, since this is usually where the primary input file for the program will be located. The TMP directory is used for temporary files which are created and deleted by the programs, and will usually be named with the extension ".TMP." The only time these files may be present after execution of the program is in the case of an abnormal abort.

In the initial distribution of the model, there are two MODELING.PRM files, one in the "root" directory, and one in the [.MODEL] sub-directory. The file in the "root" directory contains the following

TABLE B-2. DIRECTORY CODES AND FILE TYPES WHICH CAN BE SPECIFIED IN THE MODELING.PRM DATA SET, AND THE PROGRAMS WHICH USE THEM.

Code	File types	Input/output
PREPARATION PROGRAM:		
MOD	Model data (.MOD file)	Output
PHF	Photolysis data (.PHF file)	Input
LOG	Primary input file (.PRP)	Input
	Model listing file (.PRO)	Output
TMP	Miscellaneous temporary files	Input/Output
INTEGRATION PROGRAM:		
MOD	Model data (.MOD file)	Input
	Default lumping and translation file (.LMP) (if not specified explicitly)	
LOG	Primary input file (.INT)	Input
	Output listing file (.OUT)	Output
	Calculation log file (CALC.LOG)	Output
CAL	Calculated data (.CDT)	Output
	Calculated rate data (.RDT)	Output
	Calculation number file (CALC.PRM)	Input/Output
SDR	Spectral distribution data (.SDR)	Input
LMP	Kinetic and mechanistic parameter files	Input
TMP	Miscellaneous temporary files	Input/Output
PLOTTING PROGRAM:		
LOG	Primary input file (.PLT)	Input
	Output file (.PLO)	Output
CAL	Calculated data (.CDT)	Input
	Calculated rate data (.RDT)	Input
	Calculation number file (CALC.PRM)	Input
GDT	General data	Input
IMG	Experimental chamber data	Input
TMP	Miscellaneous temporary files	Input/Output
USER INTERFACE PROGRAM:		
CAL	Calculated data (.CDT)	Input
	Calculation number file (CALC.PRM)	Input
GDT	General data	Input
IMG	Experimental chamber data	Input
LOG	Written to AFJOB.COM file	Output

entries, to indicate that the .MOD files and the lumping files are located in the [.MODEL] subdirectory:

```
MOD=[.MODEL]
LMP=[.MODEL]
```

The file in the [.MODEL] subdirectory contains no entries, since all the files used when the Model Preparation program is run will be in the same directory in which that program is to be run.

B. INSTALLATION AND FILES ON CYBER SYSTEMS

1. List of Files

The data files included in the distribution of this model for Cyber systems are listed in Table B-3. This listing also includes the files which are created in the process of installing the model or running the example calculations discussed in Section V. A brief description of these files is also included in the table. All of these are located in a single user directory. Except as otherwise noted in Table B-3, all files are in direct access permanent files. NOTE: This modeling system requires that the limit for the size of any one indirect file be at least 512 PRU's. In addition, it is recommended that the limit for the number of permanent files allowed in your catalog be at least 512 and that the limit for the size of any one direct file be as large as possible.

TABLE B-3. LIST OF FILES REQUIRED FOR THE INSTALLATION OF THE MODEL AND EXAMPLES ON CYBER SYSTEMS.

A. Files Required for the Routine Use of this Model

1. Data Files Required and Maintained by the User Interface.

ALLSCEN	Contains scenario names.
ALLCOMP	Contains composition file names.
SDname	Contains data defining scenarios, where "name" is the scenario name. The initial distribution includes the following SD files, but more can be created by the user. SDEKMA1 SDEKMA2 SDEKMA3 SDEKM1A SDEKM1B
CFname	Contains composition file data, where "name" is the composition file name. The initial distribution contains the following CF files, but more can be created by the user. CFFUEL1 CFFUEL2 CFFUEL3 CFFUL1F CFFUL1T CFFUL1P CFEKMAA CFSAIS CFSYNEX
SBMTPRM	Contains highest calculation number prepared by the CALC command of the User Interface program.
SPECIES	Contains list of names of species which can be included in composition files. Also gives the number of carbons for emitted ROG species and comments describing the types of chemicals each species is used to represent. Note that the data in this file is model dependent, and that adding new species to this model requires more than just adding new names to this file.
MODELS	Contains model names which can be used with these files. If any models are to be added, the name of the new model must be added to this file using the system editor.
HELP	Text file of HELP messages for the User Interface program. This file can be edited to modify the help messages.

TABLE B-3. LIST OF FILES REQUIRED FOR THE INSTALLATION OF THE MODEL AND EXAMPLES ON CYBER SYSTEMS (CONTINUED).

EXPLAIN	Explanatory text about the User Interface program. This file can be edited to modify the explanatory text. Displayed when the MAIN mode EXPL command is given.

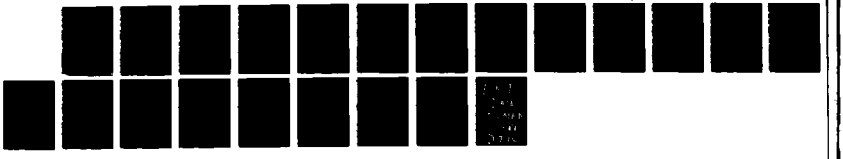
2. Data Files Maintained or Produced by the Integration or Plotting Programs.

CnnnnCD	Binary output file from the Integration program containing calculated concentration-time data from calculation number nnnn. Used as input to the Plotting program or the PLOT mode of the User Interface. The initial distribution contains at least 64 such files, consisting of results of the calculations in the examples discussed in Section V.B.
CITnnnn	Integration program listing output file for calculation group nnnn, where nnnn is the calculation number for the last calculation produced as a result of a single User Interface CALC command. The initial distribution will contain at least one such file created by one of the example calculations discussed in Section V.B, but will not contain them from all of the example calculations because of the amount of disk space they require.
PLOnnnn	Output file from Plotting program for calculation group nnnn, where nnnn is the calculation number for the last calculation produced as a result of a single User Interface CALC command. The initial distribution contains at least 16 such files, consisting of results of the calculations in the examples discussed in Section V.B.
CALCLOG	Log of all calculations run. Consists of a list of the calculation number, label, model name, and date for all calculations run. In the initial distribution, it will contain entries for the example calculations discussed in Section V.B, and for any test calculations carried out when installing the model.
CALCPRM	Contains calculation number of last calculation processed by the Integration program.

AD-A198 691 ATMOSPHERIC PHOTOCHEMICAL MODELING OF TURBINE ENGINE FUELS AND EXHAUSTS P (U) CALIFORNIA UNIV RIVERSIDE STATEWIDE AIR POLLUTION RESEARCH CE

3/3

UNCLASSIFIED W O CARTER ET AL MAY 88 F/G 4/1 NL



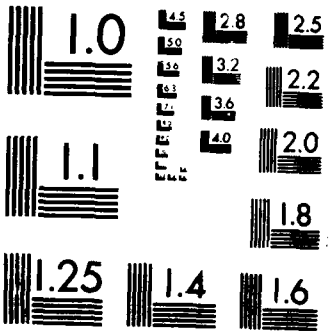

1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
6.3
7.1

TABLE B-3. LIST OF FILES REQUIRED FOR THE INSTALLATION OF THE MODEL AND EXAMPLES ON CYBER SYSTEMS (CONTINUED).

3. Temporary Files Created by the User Interface For Calculations Which Are Deleted When Calculations are Completed. (These files are not included in the initial distribution.)

AFJOB	Command file for calculations created by CALC commands during a session at the User Interface. Deleted when the calculations are completed.
INTnnnn	Integration program input file for calculation group nnnn, where nnnn is the calculation number for the last calculation produced as a result of a single User Interface CALC command.
PLTnnnn	Plotting program input file for calculation group nnnn, where nnnn is the calculation number for the last calculation produced as a result of a single User Interface CALC command.

4. Command Files and Submit Job File.

RUNSCEN	Command file used to run User Interface program. Invokes EXCALCS to automatically submit or execute calculations created by the CALC commands.
EXCALCS	Command file invoked by RUNSCEN to actually submit or run the calculations. If the calculations are to be run in batch mode, submits the job file INTJOB; otherwise, invokes the command file AFJOB.
INTJOB	Job file which can be submitted for batch processing to invoke the AFJOB command file. NOTE: the parameters for the Job command (second line in this file), may require modification to conform to the conventions of your particular computer installation. Check with your system programmers if difficulties arise trying to submit calculations for batch processing.

5. Executable Files. (These are not initially provided, but are produced during installation.)

XAF	The User Interface program. (This will be a direct access permanent file.)
XPLT	The Plotting program

TABLE B-3. LIST OF FILES REQUIRED FOR THE INSTALLATION OF THE MODEL AND EXAMPLES ON CYBER SYSTEMS (CONTINUED).

Xmodel	The Integration program for the specific model, where "model" is the name of the chemical model. (These willo be direct access permanent files.) If all the chemical models in the initial distribution of this model are prepared, this would include the following: XLPARM XM1ARM XM2ARM XLPBRM

B. Model Implementation and Preparation Files

1. Preparation Program Input Files for the Distributed Chemical Models.

PRname	Main Preparation Program input files for each of the chemical "models" which are implemented, where "name" is the model name, as referenced by the MOD scenario parameter. The initial distribution of this model includes four different chemical models based on differing ROG lumping techniques, and the preparation input files for each of these are as follows: PRLPARM PRM1ARM PRM2ARM PRLPBRM
Rname	Preparation program input files for the reactions of the various chemical species in the model. These are used by the .PRP files listed above. The .RXN files included in the initial distribution are as follows: Inorganic Reactions. Used in all models RNOXAIR Chamber-dependent reactions used when modeling chamber experiments. Used in all models. RWALLS General peroxy radical reactions. Used in all models. RRO2

TABLE B-3. LIST OF FILES REQUIRED FOR THE INSTALLATION OF THE MODEL AND EXAMPLES ON CYBER SYSTEMS (CONTINUED).

Reactions of individual organics or "lumped molecule" species and their products which are used in all models.

RHCHO	RCCHO	RRCHO	RACETON
RMEK	RBZALD	RGLYOXL	RMGLYOX
RPHENOL	RCRESOL	RRONO2	RETHENE
RFURAN	RTHIOPH	RPYRROL	

Reactions of individual organics or "lumped molecule" species used in models with LM1 or LM2 lumping.

RC4C5AL	RC6PALK	RC9PALK	RPROPEN
RT2BUT	R1HEXEN	RBENZEN	RTOLUEN
RMXYLEN	RTETRAL	RNAPHTH	RMENAPH
R23DMNA			

Reactions of "lumped molecule" species used in models with LP lumping.

RLPBZC3	RLPC4C5	RALKAR1	RALKAR2
RALKAR3	RALKAR4	RALKAR5	RLMPOL1
RLMPOL2			

2. Lumping Parameter Files. These indicate how ROG species not explicitly on the models are to be represented. Read by the Integration program, but must be consistent with the model being used. Three "master" files are included, one for each lumping approach, which must be copied into the files used by each model. The files required by the initially distributed models are as follows:

LM1LMP	Lumping file for models using LM1 lumping
LM2LMP	Lumping file for models using LM2 lumping
LPLMP	Lumping file for models using LP lumping
LPname	Lumping parameter file for model "name" which is used to convert background or aloft mix composition file data into absolute concentrations. Specific files provided:

LPLPBRM (same as LPLMP)
LPLPARM (same as LM1LMP)
LPM1ARM (same as LM1LMP)
LPM2ARM (same as LM2LMP)

TABLE B-3. LIST OF FILES REQUIRED FOR THE INSTALLATION OF THE MODEL AND EXAMPLES ON CYBER SYSTEMS (CONTINUED).

LBname	Lumping parameter file for model "name" which is used to convert data in composition files giving base case ROG emissions into emission rates and initial concentrations of model species. Specific files provided: LBLPBRM (same as LPLMP) LBLPARM (same as LM1LMP) LBM1ARM (same as LM1LMP) LBM2ARM (same as LM2LMP)
LTname	Lumping parameter file for model "name" which is used to convert data in composition files giving test mixture ROG emissions into emission rates and initial concentrations of model species. Specific files provided: LTLPARM (same as LPLMP) LTM1ARM (same as LM1LMP) LTM2ARM (same as LM2LMP)
LMALKAN	Kinetic and mechanistic parameter file for alkanes. Required by models using LP lumping and referenced by LPLMP and its copies.
LMAROMA	Kinetic and mechanistic parameter file for aromatics. Required by models using LP lumping and referenced by LPLMP and its copies.
LMALKEN	Kinetic and mechanistic parameter file for alkenes. Required by models using LP lumping and referenced by LPLMP and its copies.
ALKAPRM	Structural data for the alkanes for which LP lumping is employed. Used as input to the XALKCNV program to produce the LMALKAN file.
3. MDname	Model Data Files, where "name" is the name of the model. Not intially distributed, but produced by executing the Preparation program to prepare the distributed models for use. These are required by the Integration program to do calculations using the models. After installation of this model, the following MD files will exist. MDLPARM MDM1ARM MDM2ARM MDLPBRM

TABLE B-3. LIST OF FILES REQUIRED FOR THE INSTALLATION OF THE MODEL AND EXAMPLES ON CYBER SYSTEMS (CONTINUED).

4.	POname	Model Preparation Listing Output Files, where "name" is the name of the model. These are not required by any of the programs, and serve only to document the model. It is recommended that they be printed before they are deleted.
		POLPARM POM1ARM POM2ARM POLPBRM
5.	Executable Files. (Not initially distributed, but produced during installation.)	
	XPREP	The Model Preparation Program
	XALKCNV	The "Alkane" program, which converts alkane structural parameters in ALKAPRM to produce the mechanistic and kinetic parameters in LMALKAN.

C. Source Files and Compile and Link Command Files

1. Command (or Proc) Files

INCPROC	Compiles and links the "Include" (XINCLD) program, which is required for the compilation of all of the other programs.
CMP	Used to compile a single subroutine. Invokes the "Include" program.
PLTCL	Compiles and links the Plotting (XPLT) program.
AFCL	Compiles and links the User Interface (XAF) program.
PRPCL	Compiles and links the Model Preparation (XPREP) program.
ALKCNCL	Compiles and links the Alkane (XALKCNV) program.
LIBPROC	Compiles the Integration program modules required for linking Integration program for specific models. Compiled modules go in INTLIB file.
RUNPREP	Runs the Model Preparation program and then uses its output to compile and link the model-dependent Integration program.

TABLE B-3. LIST OF FILES REQUIRED FOR THE INSTALLATION OF THE MODEL AND EXAMPLES ON CYBER SYSTEMS (CONTINUED).

INTLNK	Links the Integration program for a specific model for which the Model Preparation has already been run. Invoked by RUNPREP.

2. FORTRAN Source Files for Utility Programs and Subroutines

FINCLUD	Source file for "Include" program, which allows specification files to be included in the source files before compilation. Required for the compilation of all programs except for this one.
FNWSUBS	Utility subroutines used by most programs.
FNWINBF	Utility subroutines controlling terminal or file input. Used by most programs.
FNWIOBF	"Include" file containing specifications of data used by NWINBF subroutines, and I/O buffers and miscellaneous specifications.

3. Source Files Specific to the User Interface (XAF) Program

FAFSPEC	FORTRAN "Include" file containing specifications for COMMON data used by the XAF program.
FDATSPC	FORTRAN "Include" file containing specifications for COMMON data used by the XAF program.
Fmodule	FORTRAN source files for program modules used in the XAF program. This does not include modules which are common both to the XAF and the XPLT program, which are given below with the source files for the other XPLT modules. The source files for the modules used only by XAF are as follows:

FAFMAN	FEDSC	FINTFIL	FPLTFIL
FCALC	FHELPSB	FLIST	FSETSCN
FCOMP	FINPSCN	FONEBY1	FDECLIN
FINPUT	FPLOTSB		

TABLE B-3. LIST OF FILES REQUIRED FOR THE INSTALLATION OF THE MODEL AND EXAMPLES ON CYBER SYSTEMS (CONTINUED).

4. Source and Documentation Files Specific to the Plotting (XPLT) Program

FPLTSPC	FORTRAN "Include" file containing specifications for COMMON data used by the XPLT program.
Fmodule	FORTRAN source files for all program modules in the XPLT program. Some of these are used by the XAF program as well. These are as follows:

FPLOT	FHELPPL	FGENDAT	FCHADAT
FCALDAT	FTABUL	FDMPDAT	FRATDAT

PLOTDOC	Draft documentation of the features of the Plotting program, including features not documented in this manual.

5. Source and Documentation Files Specific to the Model Integration Programs

INTLIB	Object code library for all the modules in Integration program which are model-independent. Not initially included, but produced by executing the LIBPROC Proc file.
FINTSPC	FORTRAN "Include" file containing specifications for COMMON data used by the Integration programs.
FLMPSPC	FORTRAN "Include" file containing specifications for COMMON data used by the lumping subroutines of the Integration program. These subroutines are LUMPHC, RDLMP, RDCMP, and WRLMO.
Fmodule	FORTRAN source files for all model-independent program modules in the Integration programs. These are as follows:

FINTMAN	FDUMOPT	FLUMPHC	FPHOTKS
FRDPHK	FGETCAL	FMODL	FPREPJ
FRDVCO	FINT	FNEWPHK	FPRNTC
FSDREAD	FINTINI	FNEWRK	FRDCKC
FSTOREC	FNEWSTP	FRDCMP	FWRLMO
FINTPRP	FPFALIN	FRDLMP	FXQKODE
FLSODE			

TABLE B-3. LIST OF FILES REQUIRED FOR THE INSTALLATION OF THE MODEL AND EXAMPLES ON CYBER SYSTEMS (CONCLUDED).

INTDOC	Draft documentation of the features and usage of the Integration program, which are not documented in this manual.

6. Source and Documentation Files Specific to the Model Preparation Program

FPSPECS	FORTRAN "Include" file containing specifications for COMMON data used by the XPREP program.
Fmodule	FORTRAN source files for all program modules in the XPREP program. These are as follows:

FASMD	FLISTSC	FRDRXN	FBLDUP
FPNHRXN	FREORDR	FDIFFUN	FPREP
FRXLST1	FINSCOD	FSAVERX	

PREPDOC	Draft documentation of the features and usage of the Preparation program, which are not documented in this manual.

7. Files Specific to the Alkane (XAKLCNV) Program

FALKSPC	FORTRAN "Include" file containing specifications for COMMON data used by the XALKCNV program.
Fmodule	FORTRAN source files for all program modules in the XALKCNV program. These are as follows:

FALKCNV	FNEWSTR	FALKANE	FSUMTYP
FISOM			

2. Compilation of Programs and Modules Required by the Model

a. Compilation of Utility Modules and Programs

The first step in compiling and linking the programs and modules required by this model is to create the XINCLUD program and to compile the utility subroutines. The XINCLUD program must exist before

most of the modules in these programs can be compiled, because it is required for the utilization of the "Include" files employed by almost all of these modules. "Include" files are used to give the specifications of global variables and data in COMMON blocks. This program compensates for the lack of an INCLUDE feature on the version of FORTRAN on the Cyber system on which this model was developed. The utility subroutines in FNWSUBS can be compiled at the same time because unlike most modules they do not require the XINCLUD program to be compiled.

To compile the source file for XINCLUD and the FNWSUBS subroutines, the following commands are given:

```
GET,FINCLUD,FNWSUBS
FTN5, I=FINCLUD,B=BINCLUD,L=LINCLUD
FTN5, I=FNWSUBS,B=BNWSUBS,L=LNWSUBS
REPLACE,BINCLUD,BNWSUBS
```

Then the XINCLUD program is linked by invoking the procedure INCPROC, by giving the commands:

```
GET, INCPROC
BEGIN,,INCPROC
```

The contents of the INCPROC procedure is as follows:

```
.PROC,INCPROC.
GET,BNWSUBS,BINCLUD.
REWIND,*.
RETURN,XINCLUD.
COPY,BINCLUD,XINCLUD.
COPY,BNWSUBS,XINCLUD.
PACK,XINCLUD.
REPLACE,XINCLUD.
LOAD,XINCLUD.
NOGO,XLINCLD.
REVERT,NOLIST.
```

After this procedure is executed, the listing files LINCLUD and LNWSUBS, and the object file BINCLUD can be deleted. However, the object file BNWSUBS must be retained until the installation procedure is completed.

The CMP procedure file is used to compile the subroutines which "include" other source files, such as those with common variable and parameter specifications. The contents of the CMP procedure file is as follows:

```
.PROC,CMP*I,FNM"NAME OF FILE TO COMPILE (6 CHARS OR LESS)"=(*F).
GET,F_FNM.
XLINCLD,F=F_FNM.
RETURN,L_FNM,B_FNM.
FTN5,I=INCLTMP,L=L_FNM,B=B_FNM,LO=0/S,REW,EL=F.
REPLACE,L_FNM.
REPLACE,B_FNM.
RETURN,INCLTMP,B_FNM.
REVERT,NOLIST.
```

Note that each time the include program runs, i.e., each time the CMP procedure runs it, it issues the message "INCLUDE DONE" upon completion. At that point, the actual compilation begins. The above message will be seen repeatedly as the programs of this modeling system are installed.

Note also that the CMP proc creates a listing file (called Lname for source file Fname), each time it is run. In the following installation directions, the CMP proc is used repeatedly to compile source files, and will thus generate many listing files. Once the programs are tested and found to run successfully, these listing files will no longer be needed and may be deleted. It may also be desirable to delete or archive the object files generated by CMP (called Bname for source file Fname), since they are not required after the installation procedure is completed, and they can always be re-created by running the appropriate procedures.

The CMP procedure is required to compile the I/O utility subroutines which are utilized by most of the required programs. To compile these subroutines, give the commands:

```
GET,CMP
CMP,NWINBF
```

This will create the listing file LNWINBF, which can deleted, and the object file BNWINBF, which should be saved until the installation procedure is completed.

b. Compilation and Linking of the Plotting (XPLT) Program

The Plotting (XPLT) program should be compiled and linked before the User Interface (XAF) program, because the latter program uses object files created when the XPLT program is compiled. To compile and link this program, give the commands:

```
GET,PLTCL
PLTCL,N
```

This invokes the PLTCL procedure, whose contents are as follows:

```
.PROC,PLTCL*I,LINKO'LINK XPLT ONLY (Y/N)?'=(YES=Y,NO=N,Y,N).
.NOTE,RECOMPILE PLOT SUBROUTINES AND LINK.
.PROMPT,SPECIFY VALUE AND PRESS RETURN WHEN READY.
.IF,$LINKO$.EQ.$N$,COMP.
CMP,PLOT.
CMP,DMPDAT.
CMP,RATDAT.
CMP,TABUL.
CMP,CALDAT.
CMP,GENDAT.
CMP,HELPPL.
CMP,CHADAT.
.ENDIF,COMP.
GET,BPLOT,BDMPDAT,BRATDAT,BTABUL,BCALDAT.
GET,BGENDAT,BPGPLT,BHELPPL,BCHADAT.
GET,BNWINBF,BNWSUBS.
RETURN,XPLT.
COPY,BPLOT,XPLT.
COPY,BDMPDAT,XPLT.
COPY,BRATDAT,XPLT.
COPY,BTABUL,XPLT.
COPY,BCALDAT,XPLT.
COPY,BGENDAT,XPLT.
COPY,BPGPLT,XPLT.
COPY,BHELPPL,XPLT.
COPY,BCHADAT,XPLT.
COPY,BNWINBF,XPLT.
COPY,BNWSUBS,XPLT.
PACK,XPLT.
REPLACE,XPLT.
RETURN,BPLOT,BDMPDAT,BRATDAT,BTABUL,BCALDAT.
RETURN,BGENDAT,BPGPLT,BHELPPL,BCHADAT.
RETURN,BNWINBF,BNWSUBS.
REVERT,NOLIST.
```

The listing (Lname) files can be deleted if desired, but the binary files should be saved until the XAF program is successfully linked.

c. Compilation and Linking of the User Interface (XAF) Program

The User Interface (XAF) program uses some of the object files created when the Plotting (XPLT) program is compiled, so it should be prepared only after the XPLT modules have been compiled. Assuming that this is the case, then the following commands can be given to compile and link the XAF program:

```
GET,AFCL
AFCL,N
```

These commands invoke the AFCL procedure file, whose contents are as follows:

```
.PROC,AFCL*I,LINKO'LINK AF ONLY (Y/N)?'=(YES=Y,NO=N,Y,N).
.NOTE,RECOMPILE AFSCEN SUBROUTINES AND LINK.
.PROMPT,SPECIFY VALUE AND PRESS RETURN WHEN READY.
.IF,$LINKO$.EQ.$N$,COMP.
CMP,AFMAIN.
CMP,SETSCN.
CMP,INPUT.
CMP,EDSC.
CMP,LIST.
CMP,INPSCN.
CMP,ONEBY1.
CMP,COMP.
CMP,HELPSB.
CMP,DECLIN.
CMP,CALC.
CMP,PLTFIL.
CMP,INTFIL.
CMP,PLOTSB.
.ENDIF,COMP.
GET,BAFMAIN,BSETSCN,BINPUT,BEDSC,BLIST,BCOMP,BHELPSB.
GET,BINPSCN,BONEBY1.
GET,BDECLIN,BCALC,BPLTFIL,BINTFIL.
GET,BPLOTSB,BCHADAT,BCALDAT,BPGPLT,BTABUL,BDMPDAT.
GET,BNWINBF,BNWSUBS.
RETURN,XXAF,XAF.
COPY,BAFMAIN,XXAF.
COPY,BSETSCN,XXAF.
```

```
COPY,BINPUT,XXAF.
COPY,BEDSC,XXAF.
COPY,BLIST,XXAF.
COPY,BINPSCN,XXAF.
COPY,BONEBY1,XXAF.
COPY,BCOMP,XXAF.
COPY,BHELPSB,XXAF.
COPY,BDECLIN,XXAF.
COPY,BCALC,XXAF.
COPY,BPLTFIL,XXAF.
COPY,BINTFIL,XXAF.
COPY,BPLOTSB,XXAF.
COPY,BCALDAT,XXAF.
COPY,BPGPLT,XXAF.
COPY,BDMPDAT,XXAF.
COPY,BTABUL,XXAF.
COPY,BCHADAT,XXAF.
COPY,BNWINBF,XXAF.
COPY,BNWSUBS,XXAF.
PACK,XXAF.
PURGE,XAF/NA.
DEFINE,XAF/NA.
REWIND,XXAF.
COPYEI,XXAF,XAF.
RETURN,BAFMAIN,BSETSCN,BINPUT,BEDSC,BLIST,BCOMP.
RETURN,BINPSCN,BONEBY1,BHELPSB.
RETURN,BDECLIN,BCALC,BPLTFIL,BINTFIL.
RETURN,BPLOTSB,BCALDAT,BCHADAT,BPGPLT,BDMPDAT,BTABUL.
RETURN,BNWINBF,BNWSUBS,XXAF.
REVERT,NOLIST.
```

After this program is successully linked, the listing (Lname) and object (Bname) files can be deleted, if desired.

d. Compilation and Linking of the Model Preparation (XPREP) Program

The Model Preparation (XPREP) program is compiled and linked by giving the following commands, which invoke the PRPCL procedure:

```
GET,PRPCL
PRPCL,N
```

The contents of the PRPCL procedure file is as follows:

```
.PROC,PRPCL*I,LINKO'LINK XPREP ONLY (Y/N)?'=(YES=Y,NO=N,Y,N).
.NOTE,RECOMPILE PREP SUBROUTINES AND LINK.
.PROMPT,SPECIFY VALUE AND PRESS RETURN WHEN READY.
.IF,$LINKO$.EQ.$N$,COMP.
CMP,PREP.
CMP,DIFFUN.
CMP,BLDUP.
CMP,RDRXN.
CMP,REORDR.
CMP,VCOEF.
CMP,INSCOD.
CMP,SAVERX.
CMP,PNHRXN.
CMP,LISTSC.
CMP,RXLST1.
CMP,ASMD.
.ENDIF,COMP.
GET,BPREP,BDIFFUN,BBLDUP,BRDRXN,BREORDR,BINSCOD.
GET,BVCOEF,BSAVERX,BPNHRXN,BLISTSC,BRXLST1,BASMD.
GET,BNWINBF,BNWSUBS.
RETURN,XPREP.
COPY,BPREP,XPREP.
COPY,BDIFFUN,XPREP.
COPY,BBLDUP,XPREP.
COPY,BRDRXN,XPREP.
COPY,BREORDR,XPREP.
COPY,BVCOEF,XPREP.
COPY,BINSCOD,XPREP.
COPY,BSAVERX,XPREP.
COPY,BPNHRXN,XPREP.
COPY,BLISTSC,XPREP.
COPY,BRXLST1,XPREP.
COPY,BASMD,XPREP.
COPY,BNWINBF,XPREP.
COPY,BNWSUBS,XPREP.
PACK,XPREP.
REPLACE,XPREP.
RETURN,BPREP,BDIFFUN,BBLDUP,BRDRXN,BREORDR,BINSCOD.
RETURN,BVCOEF,BSAVERX,BPNHRXN,BLISTSC,BRXLST1,BASMD.
RETURN,BNWINBF,BNWSUBS.
REVERT,NOLIST.
```

The listing (Lname) and binary (Bname) files created by this procedure can be deleted after this program is successfully linked.

e. Compilation and Linking of the Alkane (XALKCNV) Program

The ALKCNV program is not required to install the initial version of this model, but it is required if it is desired to add

additional alkanes which can be lumped using the "LP" approach. To create this program, give the commands:

```
GET,ALKCNCL
ALKCNCL,N
```

This invokes the ALKCNCL procedure, whose contents are as follows:

```
.PROC,ALKCNCL*I,LINKO'LINK XALKCNV ONLY (Y/N)?'=(YES=Y,NO=N,Y,N).
.NOTE,RECOMPILE ALKANE SUBROUTINES AND LINK.
.PROMPT,SPECIFY VALUE AND PRESS RETURN WHEN READY.
.IF,$LINKO$.EQ.$N$,COMP.
CMP,ALKCNV.
CMP,ALKANE.
CMP,ISOM.
CMP,NEWSTR.
CMP,SUMTYP.
.ENDIF,COMP.
GET,BALKCNV,BALKANE,BISOM,BNEWSTR,BSUMTYP.
RETURN,XALKCNV.
COPY,BALKCNV,XALKCNV.
COPY,BALKANE,XALKCNV.
COPY,BISOM,XALKCNV.
COPY,BNEWSTR,XALKCNV.
COPY,BSUMTYP,XALKCNV.
PACK,XALKCNV.
REPLACE,XALKCNV.
RETURN,BALKCNV,BALKANE,BISOM,BNEWSTR,BSUMTYP.
REVERT,NOLIST.
```

The listing (Lname) and object (Bname) can be deleted after this program is succesfully linked.

f. Compilation of the Integration Program Modules

The "Integration program" on Cyber systems is not a single program, but a series of programs, one for each model which is implemented. This is because the Integration program uses model-specific subroutines which are created by the Model Preparation program at the time the model is prepared. Thus, the final linking of the Integration program is actually part of the model preparation process. However, a vast majority of the modules in the the Integration programs are the same

regardless of which model is being employed, and thus it is much more efficient to compile these modules and place their object code in an object library, so that they do not need to be recompiled every time a new model is prepared. In order to accomplish this, the following commands are given:

```
GET,INTCOMP
INTCOMP,Y
GET,LIBPROC
BEGIN,LIBPROC
```

These commands invoke two procedures, INTCOMP and LIBPROC. The INTCOMP procedure is used to compile all of the model-independent subroutines. The installer is given the option not to recompile the LSODE and PREPJ modules because these large subroutine do not use the Integration program "include" files, and thus, do not need to be re-compiled when they have been changed (provided, of course, that the BLSODE and BPREPJ object files still exist). The contents of the INTCOMP procedure are as follows:

```
.PROC,INTCOMP*I,CMPLSOD'RECOMPILE LSODE (Y/N)?'=(YES=Y,NO=N,Y,N,*N=N).
.NOTE, RECOMPILE INT SUBROUTINES.
CMP,INTMAN.
CMP,INT.
CMP,RDCKC.
CMP,INTPRP.
CMP,NEWSTP.
CMP,PRNTC.
CMP,NEWRK.
CMP,NEWPHK.
CMP,GETCAL.
CMP,PHOTKS.
CMP,SDREAD.
CMP,STOREC.
CMP,PFALIN.
CMP,RDPHK.
CMP,RDVCO.
CMP,DUMOPT.
CMP,INTINI.
CMP,XQKODE.
CMP,MODL.
CMP,LUMPHC.
CMP,RDLMP.
```

```
CMP,RDCMP.
CMP,WRLMO.
.IF,$CMPLSOD$.EQ.$Y$,CLSOD.
CMP,LSODE.
CMP,PREPJ.
.ENDIF,CLSOD.
```

The LIBPROC procedure is used to combine the object files created by the INTCOMP procedure into an object module pseudolibrary file called INTLIB. The contents of this procedure is as follows:

```
.PROC,LIBPROC.
.NOTE,LINK INT SUBS (NOT INCLUDING MODEL SUBS).
GET,BINTMAN,BINT,BINTPRP,BINTINI,BRDCKC,BNEWSTP,BPRNTC.
GET,BNEWRK,BNEWPHK,BGETCAL,BPHOTKS,BSDREAD,BSTOREC,BPFALIN.
GET,BRDPHK,BRDVCO,BLSODE,BXQKODE,BMODL,BDUMOPT,BPREPJ.
GET,BLUMPHC,BRDLMP,BRDCMP,BWRLMO.
GET,BNWINBF,BNWSUBS.
RETURN,INTLIB.
COPY,BINTMAN,INTLIB.
COPY,BINT,INTLIB.
COPY,BINTPRP,INTLIB.
COPY,BINTINI,INTLIB.
COPY,BRDCKC,INTLIB.
COPY,BNEWSTP,INTLIB.
COPY,BPRNTC,INTLIB.
COPY,BNEWRK,INTLIB.
COPY,BNEWPHK,INTLIB.
COPY,BGETCAL,INTLIB.
COPY,BPHOTKS,INTLIB.
COPY,BSDREAD,INTLIB.
COPY,BSTOREC,INTLIB.
COPY,BPFALIN,INTLIB.
COPY,BRDPHK,INTLIB.
COPY,BRDVCO,INTLIB.
COPY,BLSODE,INTLIB.
COPY,BPREPJ,INTLIB.
COPY,BXQKODE,INTLIB.
COPY,BMODL,INTLIB.
COPY,BLUMPHC,INTLIB.
COPY,BRDLMP,INTLIB.
COPY,BRDCMP,INTLIB.
COPY,BWRLMO,INTLIB.
COPY,BDUMOPT,INTLIB.
COPY,BNWINBF,INTLIB.
COPY,BNWSUBS,INTLIB.
PACK,INTLIB.
REWIND,*.
REPLACE,INTLIB.
RETURN,BINTMAN,BINT,BINTPRP,BINTINI,BRDCKC,BNEWSTP,BPRNTC.
RETURN,BNEWRK,BNEWPHK,BGETCAL,BPHOTKS,BSDREAD,BSTOREC,BPFALIN.
```

```
RETURN,BRDPHK,BRDVCO,BLSODE,BXQKODE,BMODL,BDUMOPT,BPREPJ.
RETURN,BLUMPHC,BRDLMP,BRDCMP,BWRLMO.
RETURN,BNWINBF,BNWSUBS.
REVERT,NOLIST.
```

The INTLIB file created by this procedure is required for the preparation of models, and should be retained. The listing (Lname) and the object (Bname) files can be deleted once the INTLIB file is successfully created. The only additional files required to successfully link a complete Integration program are the model-specific files created when the model-specific source files produced by the Preparation program are compiled. These files are produced and compiled, and the model-specific Integration program is linked, as part of the model preparation process.

3. Preparation of Models and Model-Specific Integration Programs

Before any simulations with a model can be carried out, the model must be "prepared." This involves: (1) running the model preparation program for that model, which produces the model data (MD) file for that model and produces the FORTRAN source files for the model-specific Integration program subroutines; (2) compiling the model-specific subroutines; and (3) linking the model-specific Integration program. The procedure files RUNPREP and INTLNK (invoked by RUNPREP) are provided for this purpose. To carry out the steps involved in preparing a new model, the following commands are used, where "name" is the name of the model being prepared.

```
GET,RUNPREP
RUNPREP,name
```

These commands cause the Preparation program to be run using the file "PRname" as the Preparation program input file (which in turn references a number of "R" files giving reactions for specific chemical species or groups of species in the mechanism), and ultimately results in a model-specific Integration program, "Xname" to be produced, assuming that no errors occurred. In addition, the Preparation program output files "POname" and (if no Preparation program errors are encountered) "MDname"

are produced. The "PO" file gives a listing of the model, and any error or warning messages that might have been produced due to incorrect input data. The user can print this file and then delete it if desired. The "MD" file is required to do calculations using this model, and should be kept as long as the model is to be used. It is a binary file, so it should not be printed.

As indicated above, the procedure RUNPREP is used to carry out all the tasks necessary to prepare the model from the model preparation input files. The contents of the RUNPREP procedure file is as follows:

```
.PROC,RUNPREP*I,MODNAM"NAME FOR PRP FILE (<=5 CHARS)"=(*F).
GET,XPREP.
XPREP,F=MODNAM.
COMMENT. PREP DONE.
.*
REWIND,*.
.*
COMMENT. COMPILE F_MODNAM FILE AND LINK WITH INT.
RETURN,L_MODNAM,B_MODNAM.
FTN5,I=F_MODNAM,L=L_MODNAM,B=B_MODNAM,REW.
REPLACE,B_MODNAM.
REWIND,B_MODNAM.
RETURN,F_MODNAM.
.*
COMMENT. CALL PROC TO LINK WITH INT SUBS.
GET,INTLNK.
INTLNK,MODNAM,N.
REVERT. PREPARATION PROCESS COMPLETED.
.*
.* ERROR EXIT FROM PREP OR COMPILE OR LINK
EXIT.
REVERT,ABORT.  CHECK DAYFILE OR PO_MODNAM FILE.
```

Note that RUNPREP procedure invokes the prodedure INTLNK, which is used to compile the model dependent subroutines and link the the model-dependent Integration program. The contents of this file is as follows:

```
.PROC,INTLNK*I,MF"NAME OF MODEL(4 CHARS)"=(*F),
 RECMP"RECOMPILE MODEL SUBS FILE (Y/N)?"=(*N=Y,YES=Y,NO=N,Y,N).
.NOTE,COMPILE F_MF FILE AND LINK WITH INT.
.IF,$RECMP$.EQ.$Y$,COMPIL.
GET,F_MF.
RETURN, L_MF,B_MF.
FTN5,I=F_MF,L=L_MF,B=B_MF,REW.
REPLACE,B_MF.
```

```
.ELSE,COMPIL.
GET,B_MF.
.ENDIF,COMPIL.
REWIND,*.
RETURN,XX_MF,X_MF.
GET,INTLIB.
COPY,INTLIB,XX_MF.
COPY,B_MF,XX_MF.
PACK,XX_MF.
PURGE,X_MF/NA.
DEFINE,X_MF/NA.
REWIND,XX_MF.
COPYEI,XX_MF,X_MF.
RETURN,B_MF,XX_MF.
REWIND,*.
REVERT,NOLIST.
```

Note that these procedures also create the temporary files Fname, Bname and Lname (where, again "name" is the name of the model), which consist of the source, object, and listing files for the model-specific subroutines. These should be deleted if the model-dependent Integration program is linked successfuly. The necessary commands to do this could be added to the RUNPREP or the INTLNK procedure file, if desired.

The initial distribution of this modeling system includes four models, called LPARM, LPBRM, LM1ARM, and LM2ARM, which differ in how ROG species are lumped. It is necessary to prepare only those models which are actually going to be used, but if a model is not to be prepared, its name should be removed from the MODELS data set, to prevent nonprepared models being referenced in scenarios created using the User Interface. As indicated in Table B-3, the primary input files for these models are the PR files, though the reaction files (R files) must also be present in the same directory, since these are referenced by the PR files.

(The reverse of this page is blank.)